



IDA E-PROCUREMENT PROTOCOL XML SCHEMAS INITIATIVE

E-ORDERING AND E-INVOICING PHASES

IDA working document - Version 1.3 – June 2004

Disclaimer

"The views expressed in this document are purely those of the writer and may not, in any circumstances, be interpreted as stating an official position of the European Commission.

The European Commission does not guarantee the accuracy of the information included in this study, nor does it accept any responsibility for any use thereof.

Reference to any specific products, specifications, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favouring by the European Commission.

Every care has been taken by the author to ensure that he has obtained, where necessary, permission to use any parts of manuscripts including illustrations, maps, and graphs, on which intellectual property rights already exist from the titular holder(s) of such rights or from his or their legal representative."

Copyright: European Communities 2004

TABLE OF CONTENTS

1	Introduction	5
1.1	Context	5
1.2	Scope.....	5
1.3	Methodology overview.....	5
1.4	Document structure	6
2	Business requirements	7
2.1	Scope.....	7
2.2	Definition	7
2.3	Introduction	8
2.4	Actors.....	8
2.5	Use cases	9
2.6	The Ordering process.....	10
2.7	The fulfilment process.....	12
2.8	The accounting process.....	13
3	Information Model	14
3.1	Introduction	14
3.2	General principles and assumptions	14
3.3	Generic rules	15
3.3.1	Documents and lines	15
3.3.2	References between business documents and lines.....	15
3.4	UML notation	16
3.5	Package description.....	16
3.6	The Datatypes package.....	17
3.7	The Document package	19
3.8	The Line package	21
3.9	The Item package	22
3.10	The Party package.....	24
3.11	Purchase order.....	27
3.12	PO response.....	28
3.13	PO change	29
3.14	Despatch Advice.....	31

3.15	<i>Receipt advice</i>	32
3.16	<i>Rectification advice</i>	34
3.17	<i>Invoice</i>	35
3.18	<i>Invoice response</i>	37
3.19	<i>Credit note</i>	38
3.20	<i>Remittance advice</i>	40
4	Extending the IDA e-procurement model	42
5	Gap with UBL	43
6	Gap with eHandel	51
7	Gap with OGC	56
8	Appendix 1: Issues and decisions made	63
9	Appendix 2: UML to XML schemas conversion rules	68
10	Appendix 3: XML schemas design guidelines	69
10.1	<i>Naming conventions</i>	69
10.2	<i>Namespace</i>	69
10.3	<i>Versioning</i>	69
10.4	<i>Use of Data types definition</i>	69
10.5	<i>Use of attributes</i>	69
10.6	<i>Multilingual content</i>	69
10.7	<i>Extension mechanism</i>	70
10.8	<i>Schema documentation</i>	70
10.9	<i>elementFormDefault and attributeFormDefault</i>	70
11	Appendix 4: Acronyms	71

1 INTRODUCTION

1.1 CONTEXT

IDA is a European programme using information and communication technologies to support exchange of information between administrations. Its objective is to improve Community decision-making, facilitate operation of the internal market and accelerate policy implementation.

In the public procurement area, interactive services have been identified as key to reducing borders and contributing to the enforcement of the single European Market and the competitiveness of European businesses.

Therefore, IDA has undertaken in 2003 a study in order to define and promote pan-European guidelines for data exchange in the public e-procurement domain, using common and standard data description syntax - XML schemas.

This study is in line with the new legislative framework for electronic public procurement in the European Union.

The four first phases of e-procurement that will be covered by this study are the following:

- Ordering and invoicing phases (present document);
- e-Tendering and e-awarding.

1.2 SCOPE

The scope of this document is the specification of business documents exchanged between a public sector buyer and an external supplier during the **ordering and invoicing phases**, from the phase of ordering through to remittance. It covers any type of purchase made in the public sector, ie supplies, services or works.

The specification has been greatly inspired from the OGC model developed by the Office of Government Commerce in the UK, the eHandel model developed in Norway and the UBL standard edited by OASIS. It describes business processes involved and data models using UML diagrams.

This document includes in appendix a gap analysis between the three models indicated above and the one we herein propose.

Many projects or standards (EDIFACT, X12, Rosettanet, UBL and other national, regional or industry-specific initiatives) have developed electronic order and invoice messages. All the message definitions are very similar in essence but all have significant differences. The gap is usually due to different decisions as regards recurring questions or issues that arose during the process of designing them. For this reason, a dedicated section (*8. Appendix 1: Issues and decisions made, page 63*) lists issues and decisions made. This section ensures traceability and subsequently eases understanding of the IDA e-procurement model as well as maintaining it.

Note: Although an implementation of the proposed model in the form of a set of XML schemas is also provided, the model (processes and information model) does not rely on XML. It could therefore be implemented using a different technology such as ASN1 for example.

1.3 METHODOLOGY OVERVIEW

The methodology used is based on:

- The description of the business requirements by modelling business processes and business documents exchanged between parties;

- The conversion of the UML data model into XML schemas using conversion rules.

The specification cycle, based on a bottom-up approach, relied heavily upon existing work, namely:

- The OGC model¹ developed by the Office of Government Commerce in the UK;
- The eHandel² model developed by the eHandel project led in Norway;
- The UBL (v0.7 and v1.0 beta) model edited by OASIS. UBL is used in Denmark.

UBL, new e-business standard edited by OASIS with a similar scope, was not taken as-is for the following reasons:

- Needs and requirements specific to Europe and the public sector have to be addressed;
- At the time we started the study UBL was still under development (v1.0 was not released yet);
- There is no certainty that UBL will be widely adopted.

To a lesser extent, the model, when necessary, was compared to the corresponding Rosettanet PIPs.

The *Invoice* business document has also been checked against the Finvoice DTD developed and widely used in Finland.

The final report from the CEN eInvoicing focus group and the European directive 2001/115/CE were also taken into account for the design of the Invoice business document.

Although the IDA specification is based on three existing models, which comply with their national regulations (UBL being in fact a world-wide standard), an exhaustive study against all applicable national regulations and practices has not been carried out. At European level, the forthcoming public procurement directives (March 20th 2003) and the invoicing ([2001/115/CE](#)) directives have been taken into consideration.

1.4 DOCUMENT STRUCTURE

This document is structured as follows:

- *Business Requirements*: define the business processes and identify business documents exchanged;
- *Information model*: describes the data composing business documents using UML class diagrams;
- *Gap with UBL*: summarises the differences between UBL 1.0 beta and the IDA e-procurement model
- *Gap with eHandel*: summarises the differences between the eHandel model and the IDA e-procurement model;
- *Gap with OGC*: lists the differences between the OGC model and the IDA e-procurement model;
- *Issues and decisions made*: lists questions and issues encountered as well as decisions made;
- *UML to XML schemas conversion rules*: describes the rules that enable the generation of XML schemas from the UML class diagrams;
- *XML schema design guidelines*: indicate the design rules applied in the design of XML schemas.

¹ eProcurement Functional Requirements Specification, v3.0b.

² Platform independent model activity model (dated 2002/10/15) and information model (dated 2002/10/16).
IDA e-procurement protocol

2 BUSINESS REQUIREMENTS

2.1 SCOPE

The scope of this model is the specification of business documents exchanged between a public sector buyer within Europe and an external supplier in the procurement cycle from the phase of ordering through to remittance.

It covers any type of purchase made in the public sector: supplies, services or works.

The business documents are specified for implementation as XML documents, exchanged between buyers and sellers. The business document structure is defined by an (W3C) XML schema.

Processes internal to the buyer and seller parties are not in the scope of our model.

The following aspects of procurement are beyond the scope:

- Search for Supplier;
- Sourcing;
- Communication with third parties and intermediaries such as carriers, trusted third parties service providers (for time-stamping, managing of digital certificates or archiving), banks or fiscal authorities;
- Messaging transport (SMTP, HTTP, SOAP, ebMS, etc.) and security³.

2.2 DEFINITION

A **business document** is a unit of business information exchanged in a *business transaction*.

An **(information) component** is a collection of data making a coherent whole (such as Address, Party, ContactInformation).

A **reusable component** is an information component that, although it does not correspond to a business document as such, it is (or is likely to be) used in different business documents.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC as quoted here:

1. MUST This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
2. MUST NOT This phrase, or the phrase "SHALL NOT", mean that the definition is an absolute prohibition of the specification.
3. SHOULD This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
4. SHOULD NOT This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label. Bradner Best Current Practice
5. MAY This word, or the adjective "OPTIONAL", mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides)

³ Apart from an optional signature element attached to each business document.

2.3 INTRODUCTION

This section describes the functional requirements of the eprocurement cycle as described above by defining:

- Actors involved in the different processes necessitating an exchange of data;
- Business processes described by UML diagrams.

2.4 ACTORS

The two main actors considered in our model are:

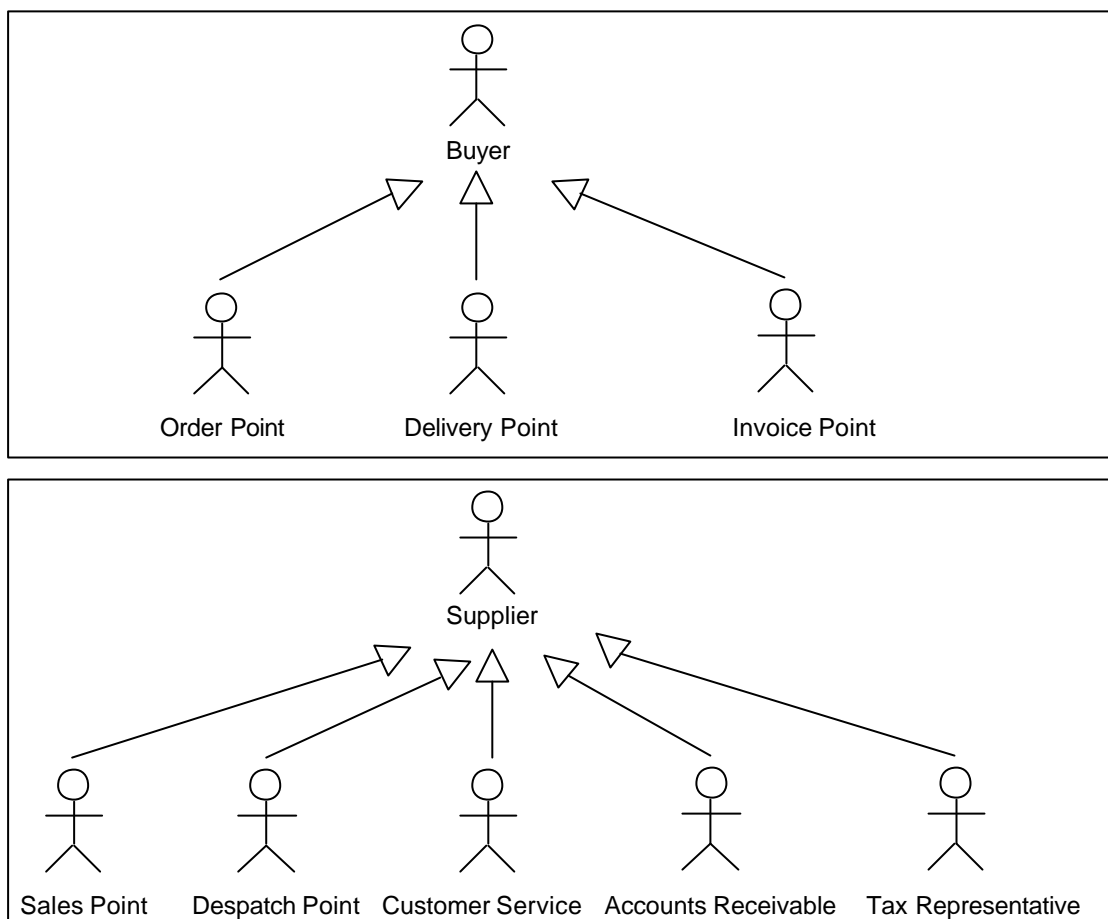
- The (public sector) buyer: the party that buys goods, services or works;
- The supplier (also called *seller*): the economic operator that supplies goods/services or works to the buyer.

All processes (and subsequent information exchange) internal to either the buyer party or the supplier party are outside the scope of our model.

Nevertheless, we have identified, for clarity purposes, particular roles within the two parties that are involved in the processes hereafter described.

These roles are as follows:

- **Buyer:**
 - **Order Point:** buyer party for issues related to purchase orders;
 - **Delivery Point:** buyer party that receives goods/works/services and identifies variances in receipt;
 - **Invoice Point:** buyer party for payment-related issues.
- **Supplier:**
 - **Sales Point:** supplier party responsible for purchasing issues prior to fulfilment of Purchase Order.
 - **Despatch Point:** supplier party responsible for the delivery of goods;
 - **Customer Service:** supplier party responsible for issues related to purchase order fulfilment (despatch, delivery and carriage);
 - **Accounts Receivable:** supplier party for payment-related issues;
 - **Tax Representative:** supplier party responsible for keeping VAT records and accounts for taxable transactions in another Member State.



UML presentation of roles

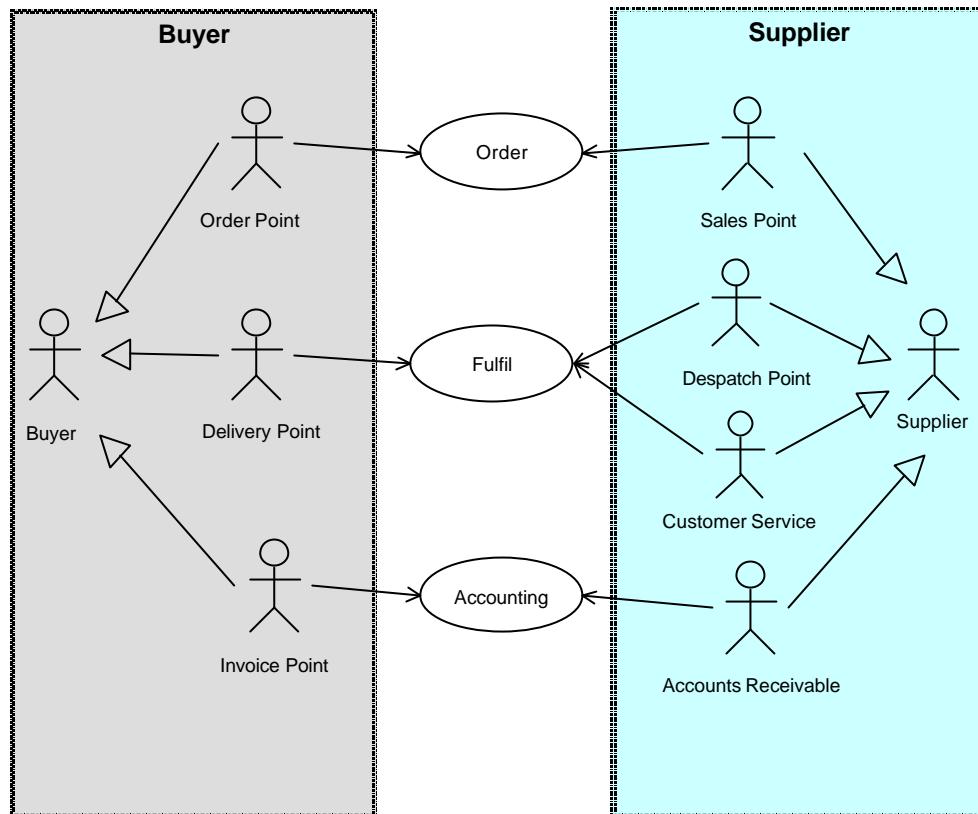
2.5 USE CASES

Use cases define specific functions that necessitate electronic data exchange covered by our model⁴.

The functions covered by our model are:

- **Ordering:** the buyer notifies a supplier of his intention to buy goods, services or works;
- **Fulfilment:** the supplier organises the delivery of the products ordered (goods, services or works) by the buyer;
- **Accounting:** the supplier sends the buyer an invoice corresponding to previously ordered and delivered goods, services or works.

⁴ Note that in our case we are not designing a software application but rather a set of messages (or business documents).



Use case diagram

2.6 THE ORDERING PROCESS

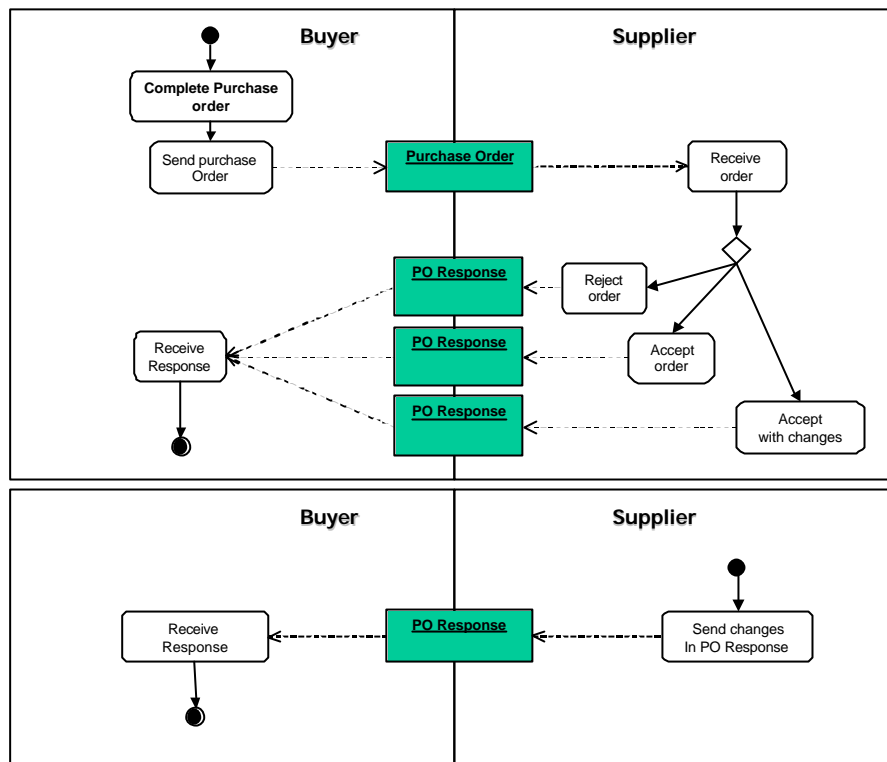
The ordering process starts with the preparation of the purchase order by the buyer.

The purchase order is sent to the supplier that receives the order.

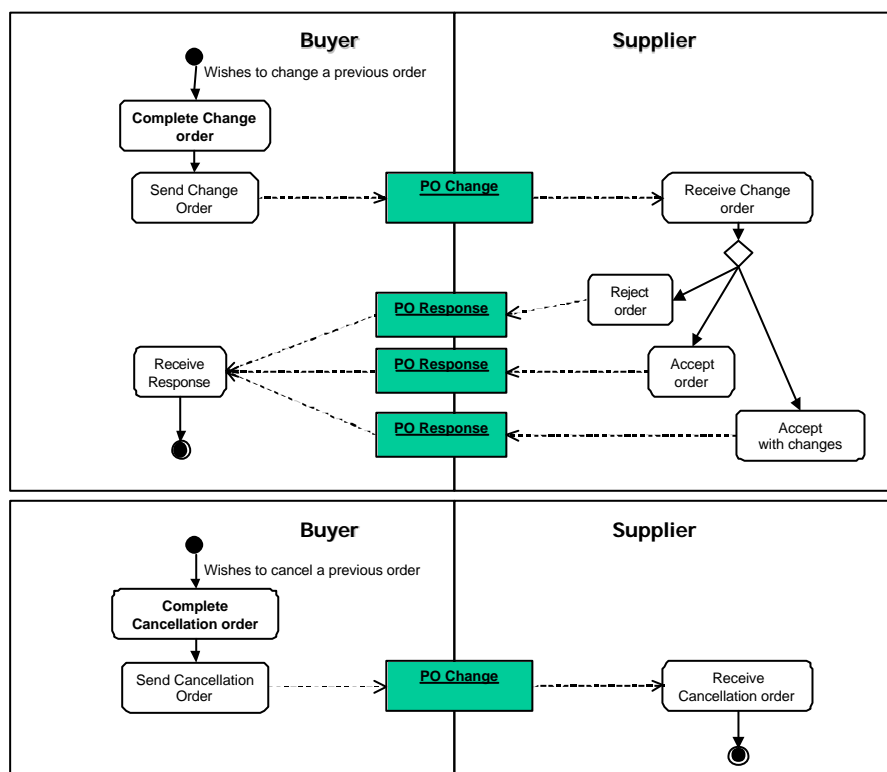
The order is accepted, rejected or accepted with changes by the supplier.

Depending on trading agreements and/or applicable legislations:

- An order may be cancelled or changed at any moment by the buyer by sending a Purchase order change document (PO Change); in this case, the supplier has to go through the process of acceptance of the order and send a Purchase order response document (PO Response) to the buyer;
- The supplier may change an order that he has already replied to (with a previous PO Response) by sending a new PO Response.



Activity diagram for the Purchase order process



Activity diagram for the Change and Cancel order process

Business document	Description
Purchase Order	Sent by the buyer to the supplier to inform him that he wishes to

IDA e-procurement protocol 11 / 73

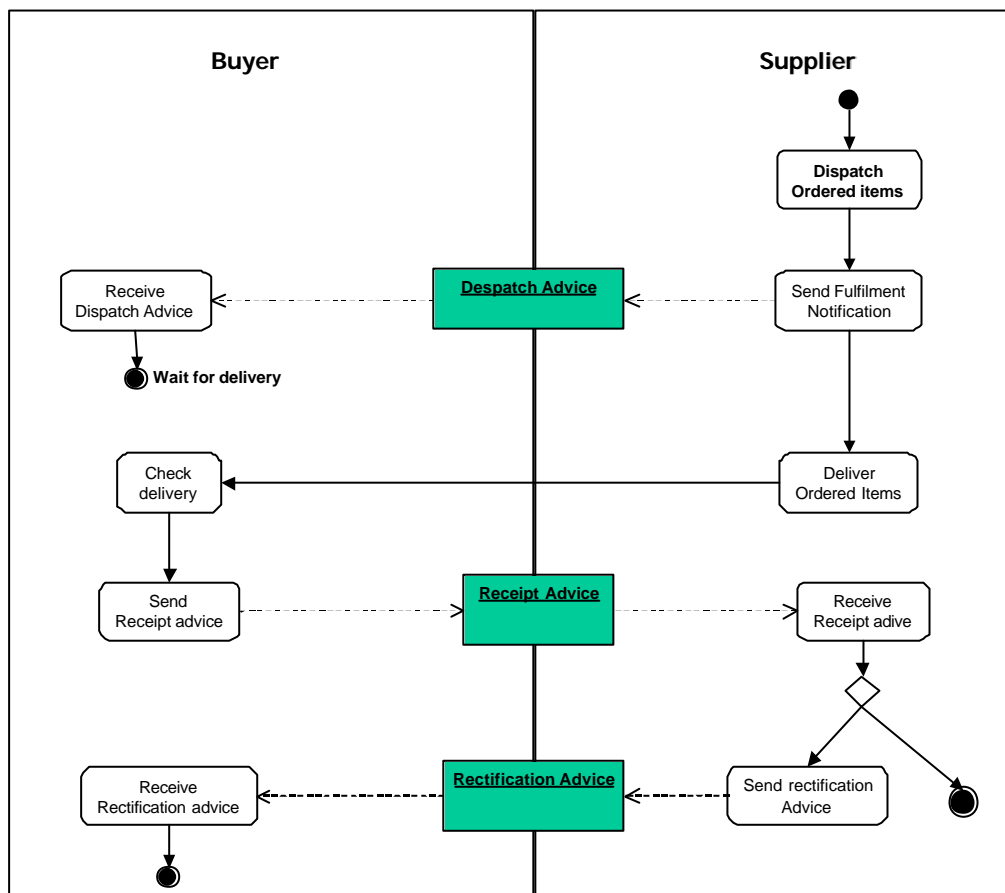
Business document	Description
	purchase goods, services or works.
PO Response (for Purchase order response)	Sent by the supplier to the buyer (who has previously sent him an order) to accept the order fully or partially or to reject the order.
PO Change (for Purchase order change)	Sent by the buyer to the supplier to inform him that he wishes to modify or cancel a previous order.

2.7 THE FULFILMENT PROCESS

The delivery process starts when the supplier dispatches the goods or services to the buyer. He sends a dispatch advice to inform the buyer.

When the goods or services are actually delivered, the buyer checks the delivery (in terms of quantity and quality) and sends a receipt advice in order to acknowledge the receipt in full or in part and notify when necessary of under/over-delivery, error or damage in the delivery.

If the receipt advice contains variances, the supplier can send a rectification advice to the buyer to inform him of the corrective action to be taken.



Activity diagram for the fulfilment process

Business document	Description
Despatch Advice	Sent by the supplier to the buyer to inform him that goods or services have been dispatched and/or delivered.
Receipt Advice	Sent by the supplier to the buyer to acknowledge receipt in full or in part and notify when necessary of under/over-delivery, error or damage in the delivery.
Rectification Advice	Sent by the supplier to inform the buyer of the action to be taken as regards as variances notified in the receipt advice.

2.8 THE ACCOUNTING PROCESS

A supplier sends an invoice to the buyer as a request for payment related to the provision of goods, services or works.

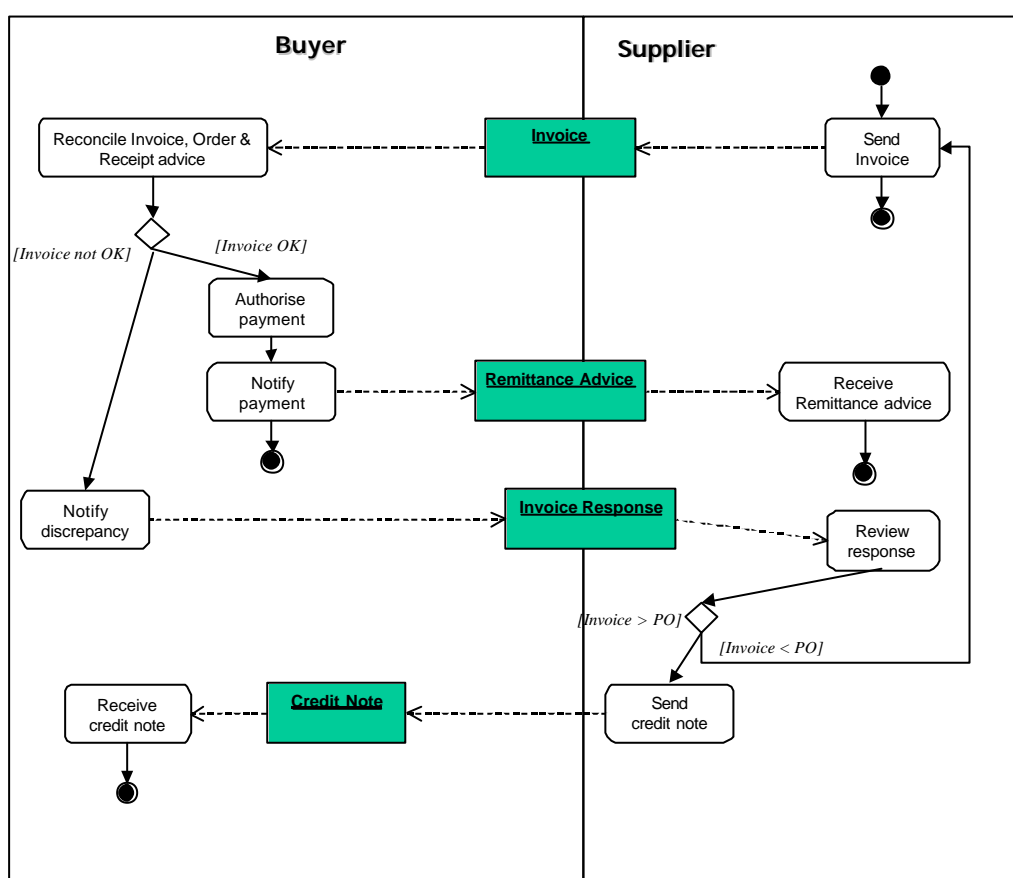
The buyer reconciles receipt advice, order and invoice and MAY either:

- Authorise payment: in that case, he notifies the payment by sending a *remittance advice* to the supplier;
- In case of discrepancy, notifies the supplier by the sending of an *Invoice Response* describing the reason.

Should the invoice amount be superior to the amount actually due, the supplier sends a *Credit note* to balance the invoice.

If the invoice amount is inferior to the amount due by the buyer, the supplier sends a new invoice for the difference.

Not that in some cases (in particular with particular payment means) not illustrated in our diagram, the invoice MAY be sent after payment is made.



Activity diagram for the accounting process

Business document	Description
Invoice	Sent by the supplier to the buyer to request for payment for goods, services or works.
Remittance Advice	Sent by the buyer to the supplier to notify payment.
Invoice Response	Sent by the supplier to the buyer in response to an invoice to inform him of discrepancies in the invoicing process.
Credit Note	Sent by the supplier to the buyer confirming that the supplier owes him money.

3 INFORMATION MODEL

3.1 INTRODUCTION

This section defines the Information Model for the different business processes that are described in the previous section.

The Information Model is composed of:

- A set of 5 reusable components, organised into packages, that are involved in the different business messages:
 - Document;
 - Line;
 - Item;
 - Party;
 - Data type;
- A Message Model that describes the 10 business documents identified in the business process description (2. *Business requirements*, page 7):
 - PurchaseOrder;
 - POResponse (PurchaseOrderResponse);
 - POChange (PurchaseOrderChange);
 - DespatchAdvice;
 - ReceiptAdvice;
 - RectificationAdvice;
 - Invoice;
 - InvoiceResponse;
 - CreditNote;
 - RemittanceAdvice.

For each UML diagram, all classes and attributes are listed and defined, with:

- The name of the attribute,
- A description of the attribute,
- The data type of the attribute,
- The cardinality of the attribute (whether it is mandatory or not):
 - 0..1 means the attribute is optional (it may have no value);
 - 1..1 means the attribute is mandatory (it must have a value).

3.2 GENERAL PRINCIPLES AND ASSUMPTIONS

Combination of paper and electronic exchanges: any business transaction can be through electronic means even if other business transactions in the same procurement cycle are carried out through paper. In some cases, a business document may also be sent by electronic means to a third party that will send a paper copy to the final recipient or vice-versa.

Simplicity. the model should be simple in order to facilitate its adoption by Small and Medium-sized Enterprises (SMES). To ensure that the IDA model remain simple enough, we decided to apply the following rule:

“In order for a piece of information to be added to the model:

- Someone should provide a well defined business case where this information is necessary and used

AND

- It should be proven that this information could be processed automatically in a reliable manner and that this represents an important benefit⁵.”

Clarity and Explicitness: other well-known B2B standards make heavy use of complex rules in order to interpret the content of a business document. For example, standards like xCBL or RosettaNet frequently

⁵ Otherwise, this information can be, when required, added to the free text notes attributes at either document or line level.

use default attributes (like currency, exchange rates, etc.) defined at document level that can be overridden at line-level. We decided that all information should be specified explicitly should it be repeated several times. The reason behind this is that this simplifies automatic processing and makes the business documents more human-readable and less ambiguous.

For clarity purposes, all business documents should as far as possible contain all pertinent information (reiteration of items ordered, parties involved, etc.). We thereby ensure that each business document is self-sufficient in interpreting it.

Traceability: many projects or standards (EDIFACT, X12, Rosettanet, UBL and other national, regional or industry-specific initiatives) have developed electronic orders and invoicing messages. All the message definitions are very similar in essence but all have significant differences. The gap is usually due to different decisions as regards recurring questions or issues that arose during the process of designing them. For this reason, a dedicated section of this document lists issues and decisions made. This section ensures traceability and subsequently eases understanding of the IDA e-procurement model as well as maintaining it.

3.3 GENERIC RULES

3.3.1 DOCUMENTS AND LINES

All information exchanged between buyer and supplier shown on the UML diagrams in 2. *Business requirements* constitute a business document.

Each of the business documents mentioned above corresponds to a class (called *Business document class*) with the same name in the *document* package.

All business documents classes are a specialisation of the *Document* class.

All business documents **MUST** have a document identifier so it can be referenced to later in the procurement cycle.

All business documents are composed of specific types of lines (class named *<BusinessDocument>Line*, i.e. *PurchaseOrderLine*, *InvoiceLine*, etc.). Each specific line class is a specialisation of the *Line* class.

All business documents **MAY** carry a digital signature.

All *Lines* **MUST** have an identifier, unique within the *Document* it belongs to, so that it can be referenced to later in the procurement cycle.

A line **MUST** be identified with a pair of attributes corresponding to a Line identifier **AND** a Document identifier.

All business document class specifies at least one buyer party (OrderPoint, DeliveryPoint, InvoicePoint) and one supplier party (SalesPoint, AccountsReceivable, CustomerService, DespatchPoint).

The “response documents” (POResponse, InvoiceResponse) repeats all the information (amended when necessary) of the business document it replies to (respectively *PurchaseOrder* and *Invoice*).

3.3.2 REFERENCES BETWEEN BUSINESS DOCUMENTS AND LINES

When a paper business document has to be referred to by an electronic one it simply specifies the identifier given on the paper document. In order, to participate efficiently in an electronic procurement cycle, a paper document has to explicitly specify a unique⁶ identifier.

When referring to a line of a paper business document, the position of the line (from the top of the document starting from 1) as it appears on paper should be used.

⁶ Unique at least within all business documents exchanged with the other party.
IDA e-procurement protocol

In an ideal case, the lines of all the business documents from *PurchaseOrder* to *Invoice* would match exactly: What was ordered (*PurchaseOrder* business document) was accepted by the supplier (*POResponse*) then delivered (*DespatchAdvice*) to the buyer, received and accepted by him (*ReceiptAdvice*) and finally invoiced (*Invoice*) by the supplier to the buyer.

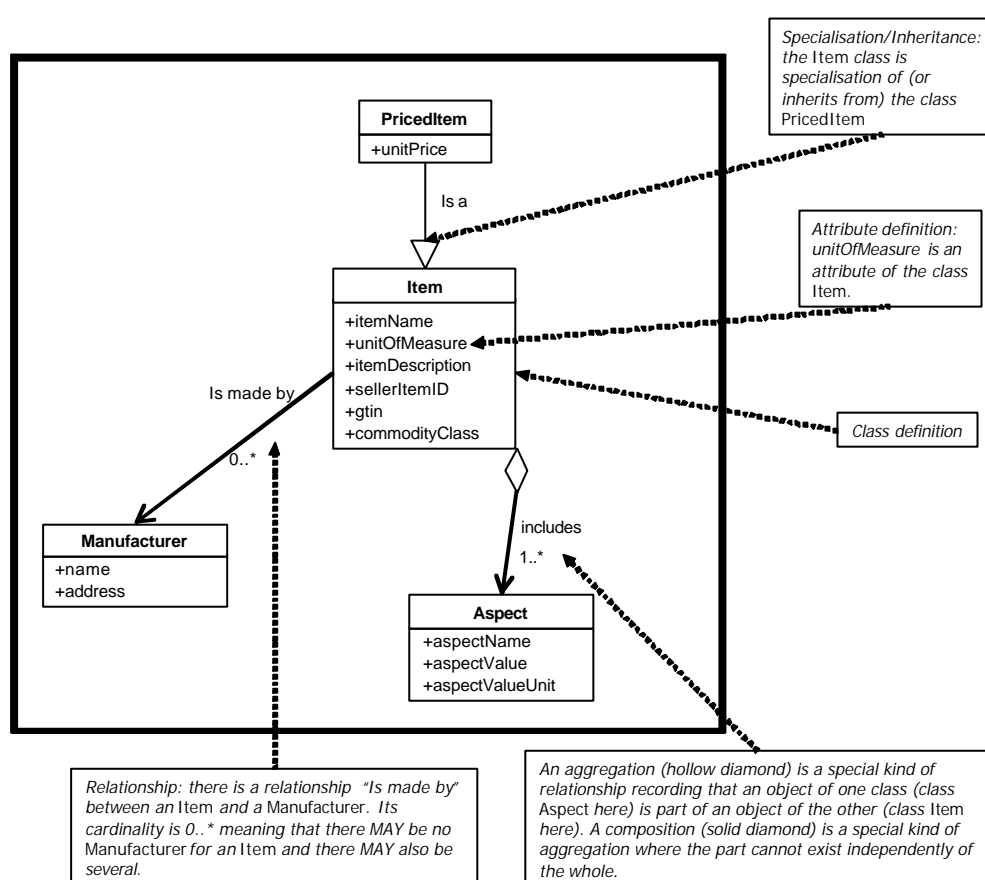
For the sake of readability and visual matching, we recommend that business documents, when referring to lines from previous business documents use the same order.

3.4 UML NOTATION

The information model specification relies on UML class diagrams showing a group of classes and the relationships between them.

To understand this specification, it is therefore necessary to have a basic knowledge of the concepts and notation used in UML and in particular in class diagrams.

The following diagram serves as a legend for all the class diagrams given hereafter:

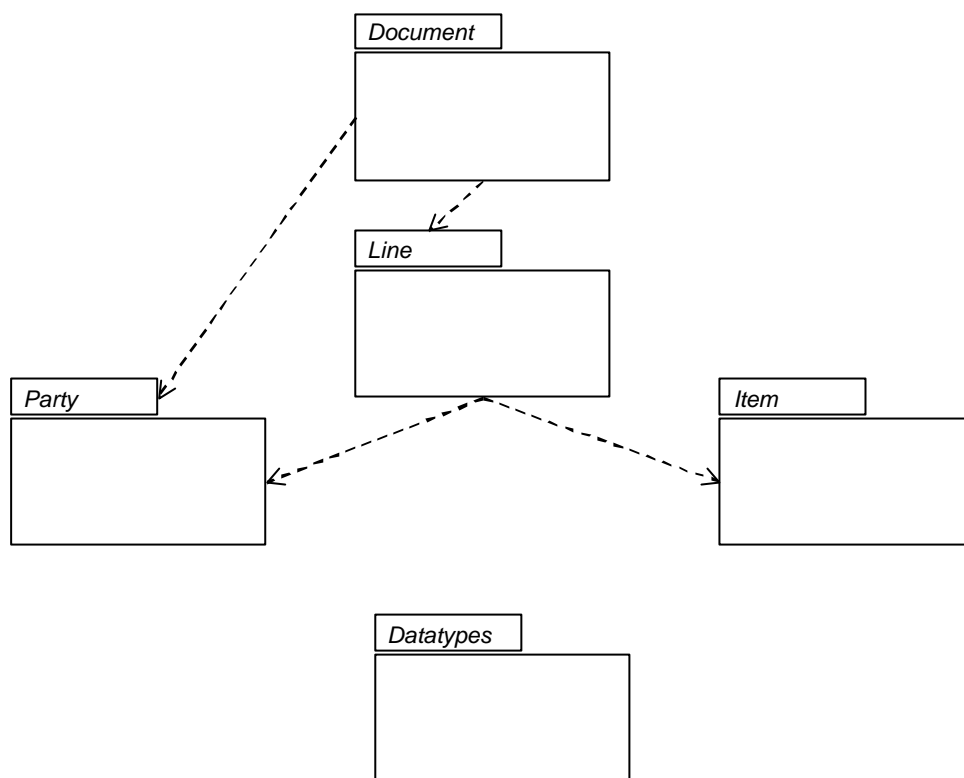


3.5 PACKAGE DESCRIPTION

The model is organised into 5 different packages:

- **Document** package, containing general information related to any document as well as pricing, charges, discounts and taxes;
- **Line** package, containing general information related to any line as well as pricing and exchange rate data;
- **Item** package, containing general information about sold items;
- **Party** package, containing information (party and contact person details, addresses and bank data) concerning each party;
- **Datatypes** package, containing the data types used throughout the whole model.

The package diagram of the global UML model is illustrated below. Arrows designate the references between packages. The *Datatypes* package is used by all other packages⁷.



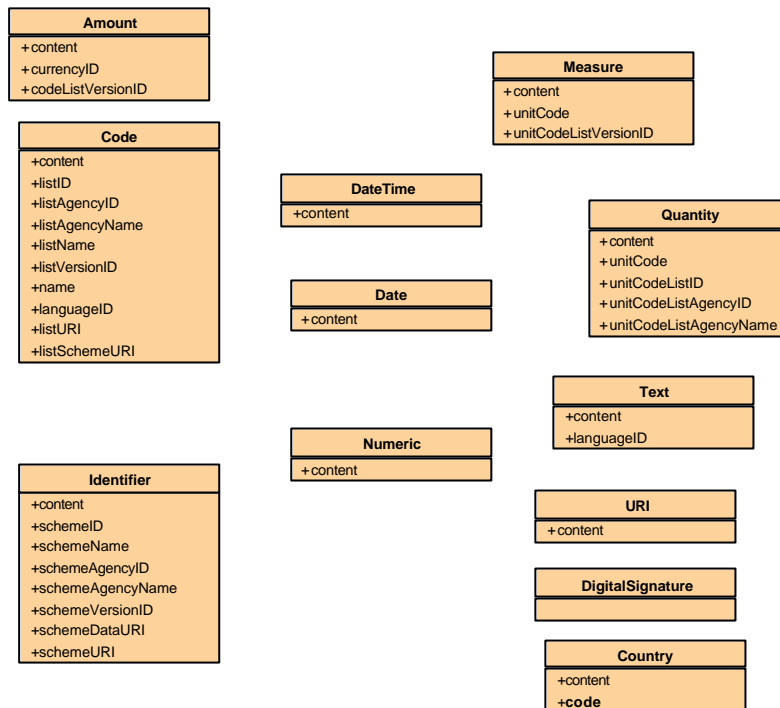
3.6 THE DATATYPES PACKAGE

Data types correspond to UBL/Core components data types and representation terms. They have been completed with other ones (URI, DigitalSignature and CountryCode).

The following picture shows⁸ the different data types used in the model:

⁷ In order to simplify the package diagram, arrows between all the packages to the *Datatypes* package are not shown.

⁸ The definition of the different attributes is given in the ebXML Core Components technical specifications.
IDA e-procurement protocol



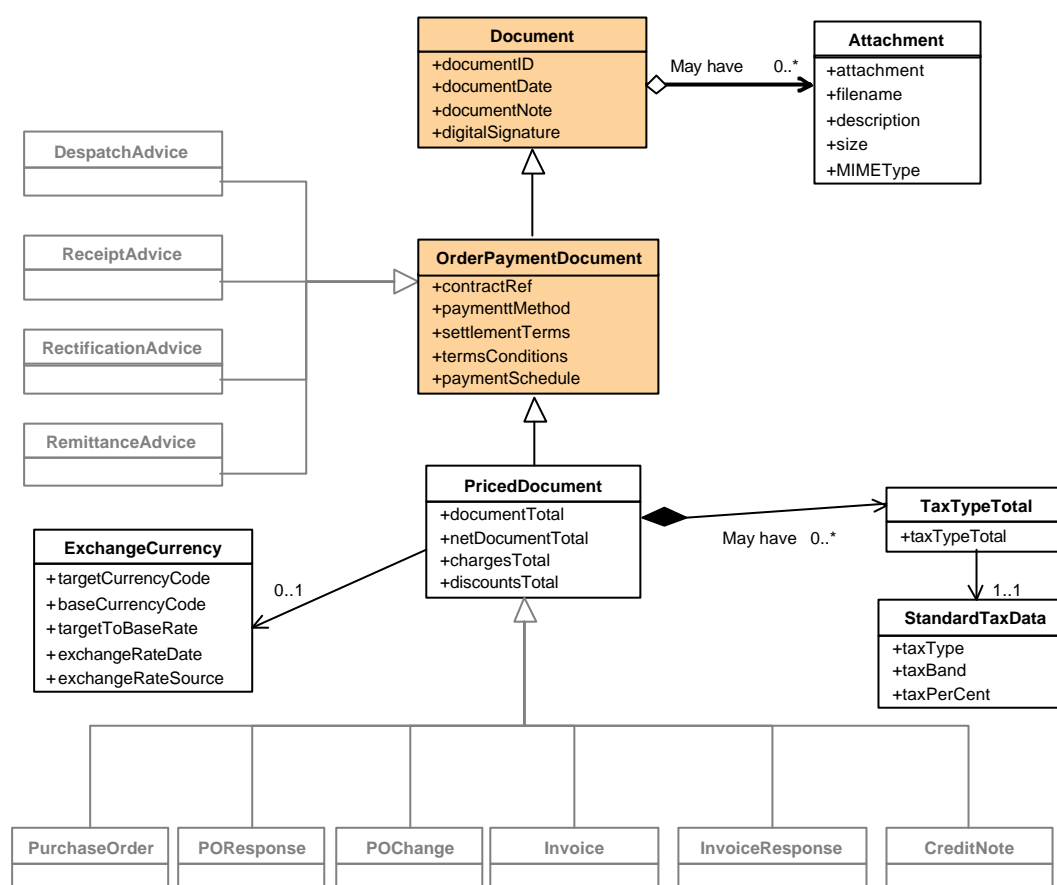
Data type	Description
Amount	A number of monetary units specified in a currency where the unit of the currency is explicit or implied
Code	A character string (letters, figures, or symbols) that for brevity and/or language independence may be used to represent or replace a definitive value or text of an attribute together with relevant supplementary information.
DateTime	A particular point in the progression of date & time together with the relevant supplementary information
Date	A particular point in the progression of date together with the relevant supplementary information
Identifier	A character string to identify and distinguish uniquely, one instance of an object in an identification scheme from all other objects in the same scheme together with relevant supplementary information
Measure	A numeric value determined by measuring an object along with the specified unit of measure
Numeric	Numeric information that is assigned or is determined by calculation, counting, or sequencing. It does not require a unit of quantity or unit of measure
Quantity	A counted number of non-monetary units possibly including fractions along with the specified unit of quantity
Text	A character string (i.e. a finite set of characters) generally in the form of words of a language
URI	Uniform Resource Identifier
DigitalSignature	A digital signature of a document. Content correspond to W3C XML Signature.
Country	The ISO country code along with the optional name of the related country

International code lists used in the model:

Code type	International code list
Country	ISO 3166
Currency	ISO 4217
Delivery terms	UN/EDIFACT No 5
Language	ISO 639
Payment method	UN/EDIFACT No 17
Unit of Measure	UN/EDIFACT No 20
Commodity class	UNSPSC, CPV, eCI@ss

3.7 THE DOCUMENT PACKAGE

The **Document Package** UML class diagram is illustrated below.



A specific section is dedicated to the description of each business document class.

Attribute	Description	Type	Cardinality
Document			
This class contains general information which allows identifying documents			
documentID	The document identifier in the sender's system	Identifier	1..1
documentDate	Date & time when the document has been issued	DateTime	1..1
documentNote	Free text note on document	Text	0..1
digitalSignature	The digital signature for the whole of the document payload	Signature	0..1
OrderPaymentDocument			
This class is a specialisation of Document that contains common data used in e-procurement processes			

Attribute	Description	Type	Cardinality
contractReference	The reference to the contract that governs the transaction	Text	0..1
paymentMethod	The method that will be used for payment. Values list: Cheque, Direct debit, Standing Order, BACS, SWIFT, Promissory note, Letter of credit, Cash, Cash on delivery, Credit card, Debit card, Charge card, Pre-paid	Code	0..1
settlementTerms	Free text that describes the terms of the document for the payment of the invoice	Text	0..1
termsConditions	Free text (or even a URL) that describes the terms & conditions (excluding delivery terms) related to the document	Text	0..1
paymentSchedule	Free text describing the payment schedule of the transaction	Text	0..1
PricedDocument			
This class is a specialisation of Document that contains financial data relating to document including gross and net total, charges and discount amounts			
documentTotal	The total of the document, including charges, discounts and taxes when appropriate	Amount	1..1
netDocumentTotal	The net total of the document, before charges, discounts and taxes when appropriate	Amount	1..1
chargesTotal	The total amount for all charges, if any	Amount	0..1
discountsTotal	The total amount for all discounts	Amount	0..1
Attachment			
This class allows attaching files (images, PDFs, ...) or URL to a document or a line			
attachment	Attached files or URLs	URI	1..1
description	Free text describing the purpose of the attachments to which it refers	Text	0..1
filename	File name of the attached file	Text	1..1
MIMEType	MIME type of the attached file	Code	0..1
size	Size of the related attachment	Measure	0..1
TaxTypeTotal			
This class contains the total amount for a specific tax type of a document			
taxTypeTotal	Total amount for a specific tax type (at specified band or percentage)	Amount	1..1
StandardTaxData			
This class contains the detailed information when considering tax of a document or of a line			
taxType	Type of tax (e.g. VAT). Mandatory if taxTypeTotal exists	Code	1..1
taxBand	Band of tax. Values list: Standard, Zero-rated, Exempt, Reduced	Code	0..1
taxPerCent	Percentage of tax	Percent	1..1
ExchangeCurrency			
This class contains information relating to exchange currency used at Document level			
baseCurrencyCode	The operational and original currency ISO code of the document	Code	1..1
targetCurrencyCode	The target currency ISO code of the document	Code	1..1
targetToBaseRate	The exchange rate that is used	Rate	1..1
exchangeRateDate	The date upon which the exchange rate is set	DateTime	0..1
exchangeRateSource	The source of the exchange rate data	Text	0..1

3.8 THE LINE PACKAGE

Lines are components of a document (which may contain one or several ones).

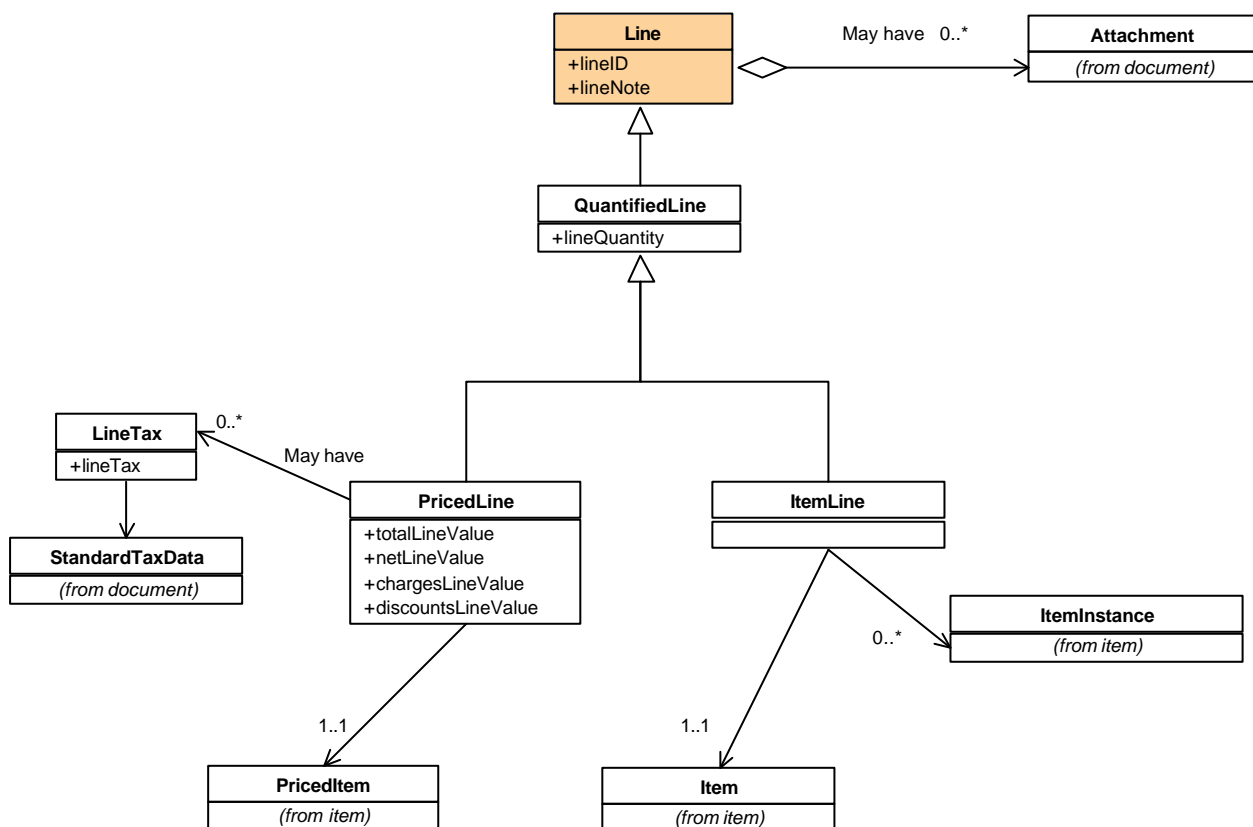
A business document, except for *RemittanceAdvice*, is composed of either *PricedLine* or *ItemLine*.

A *PricedLine* corresponds to a quantity of only one *Item* indicating all relevant pricing information (including taxes, charges and discounts).

ItemLines are used by business documents related to the delivery process (*DespatchAdvice*, *ReceiptAdvice*, *RectificationAdvice*). They correspond to a quantity of only one *Item* (without mentioning pricing information) and may specify information (serial number, batch number, etc.) related to specific *ItemInstances*.

Note that sub lines are not supported.

The **Line Package** UML class diagram is illustrated below.



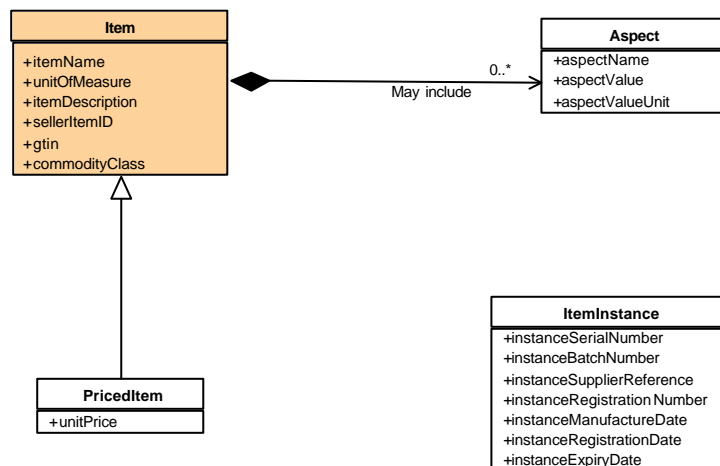
Attribute	Description	Type	Cardinality
Line			
This class contains general information allowing identifying and describing a line			
lineID	The line identifier or the line number within a document	Identifier	1..1
lineNote	Free text note on line	Text	0..1
QuantifiedLine			
This class is a specialisation of Line that contains a line quantity for the Item			

Attribute	Description	Type	Cardinality
lineQuantity	Number of item units of the line	Quantity	1..1
PricedLine			
This class is a specialisation of Line that contains financial information including gross and net total amounts			
totalLineValue	The total of the line including charges, discounts and taxes when appropriate	Amount	1..1
netLineValue	The net total of the line before charges, discounts and taxes when appropriate (unit price * line quantity)	Amount	1..1
chargesLineValue	The amount for all charges for the line, if any	Amount	0..1
discountsLineValue	The total discount for the line, if any	Amount	0..1
LineTax			
This class contains the total amount for tax of a line			
lineTax	Total amount of tax to be charged on the line	Amount	1..1
ItemLine			
This class is a specialisation of Line that may relate to item instance information			

3.9 THE ITEM PACKAGE

Items are composed of information that allows identifying (*itemName*, *sellerItemID*, *gtin*, *commodityClass*) and describe (*itemDescription*, *Aspect* class) the goods, products, items or services to procure.

The **Item Package** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
Item			
This class contains general information allowing a specific item			
itemName	The name of the good or service	Text	1..1

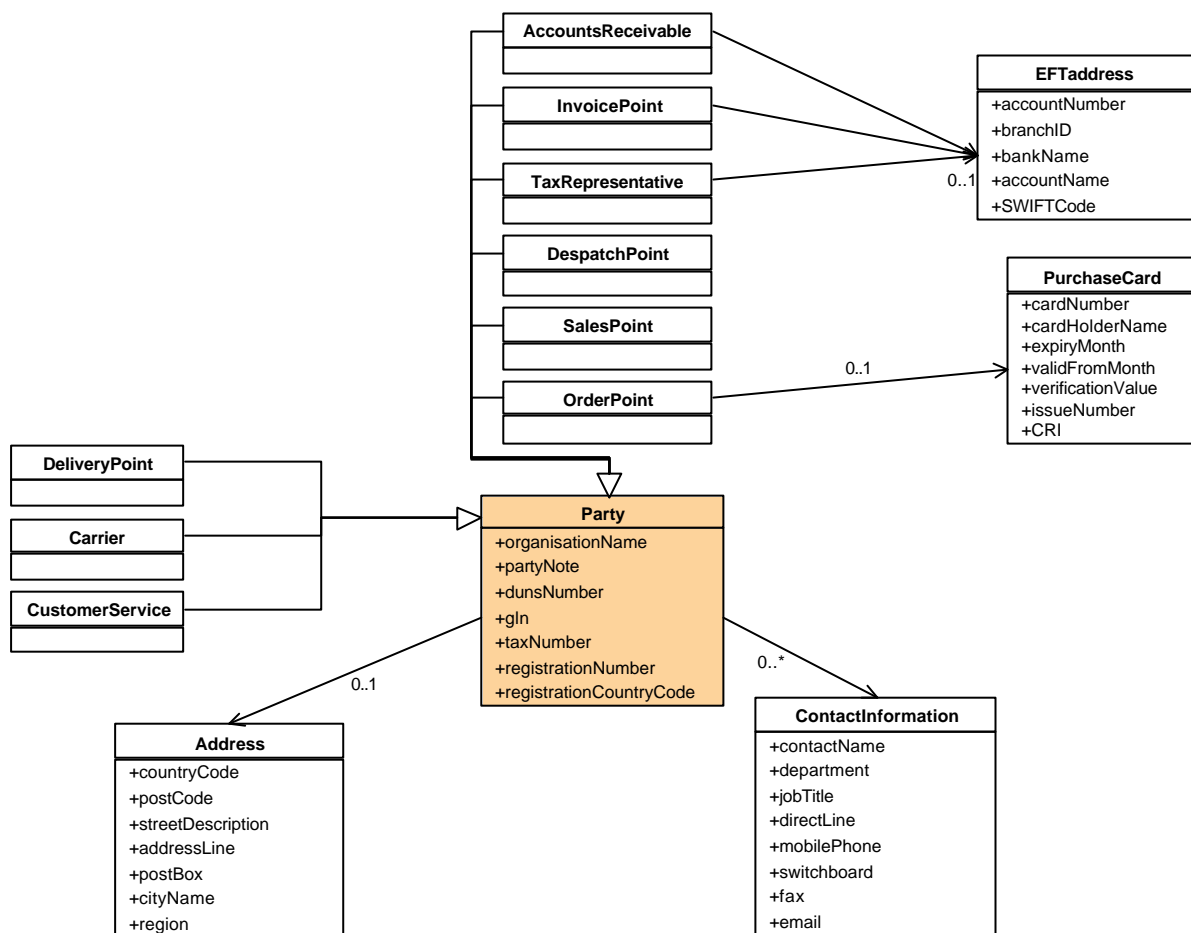
Attribute	Description	Type	Cardinality
unitOfMeasure	A UN/EDIFACT No 20 code for the quantity in which items are priced.	Code	1..1
itemDescription	Free text for a complete and full description of the item	Text	0..1
sellerItemID	The identifier for the item (from the seller form)	Identifier	0..1
gtin	The Global trade Identification Number (EAN/UCC) for the item	Text	0..1
commodityClass	The commodity class of the item using UNSPSC, CPV, eCl@ss or any other classification	Code	0..1
PricedItem			
This class is a specialisation of Item that contains financial information including the unit price of an item			
unitPrice	The unit price of the item (including the currency code)	Amount	1..1
Aspect			
This class allows providing additional information on the characteristics of an item			
aspectName	Type of the item aspect (e.g. colour, height, length width, weight, collar size)	Text	1..1
aspectValue	Value of the aspect type that has been indicated	Text	1..1
aspectValueUnit	The unit code relating to the item aspect value	Code	0..1
ItemInstance			
This class contains detailed information of a particular instance of an item of a line			
instanceSerialNumber	Product serial number	Text	0..1
instanceBatchNumber	The manufacturer's batch number	Text	0..1
instanceSupplierReference	The supplier's own reference of the item instance	Text	0..1
instanceRegistrationNumber	The registration number of the item instance	Text	0..1
instanceManufactureDate	The date of manufacture of the item instance	DateTime	0..1
instanceResgistrationDate	The date of registration for the item instance	DateTime	0..1
instanceExpiryDate	The expiry date of the item instance	DateTime	0..1

3.10 THE PARTY PACKAGE

The Party class manages information on the organisation and people involved in a Document.

Parties are composed of general information that allows identifying them. They also contain contact people details information as well as postal and electronic addresses. Finally they may include bank information such as the Electronic Funds Transfer address of the Accounts receivable or the Invoice Point parties and the Purchase Card information of the Order Point (required to process the payment).

The **Party Package** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
Party			
This class contains general information which allows identifying a party			
organisationName	The name of the organisation sending, receiving or referenced in any document	Text	1..1
partyNote	Free text that describes the party (including map or image of the location)	Text	0..1
dunsNumber	An identifier assigned to a company in the D&B Company Register	Text	0..1
gln	A Global Location Number assigned to an organisation by the EAN.UCC	Text	0..1
taxNumber	The VAT number assigned to an organisation registered for tax by the relevant national authority	Text	1..1
registrationNumber	The company's registration number	Text	0..1

Attribute	Description	Type	Cardinality
registrationCountryCode	The country where the organisation is registered	Code	0..1
ContactInformation			
This class contains information which provides contact details relating to a party			
contactName	Name of the contact person of the organisation	Text	1..1
department	Name of the department to which the contact person belongs	Text	0..1
jobTitle	Job title of the contact person	Text	0..1
directLine	Direct line of the contact person of the organisation	Text	0..1
mobilePhone	Mobile phone number of the contact person of the organisation	Text	0..1
switchboard	Switchboard of the organisation	Text	0..1
fax	Fax number of the contact person of the organisation	Text	0..1
email	Email address of the contact person of the organisation	Text	0..1
Address			
This class provides information relating to the postal address of a party			
countryCode	Country code of the organisation	Country	1..1
postCode	Post code of the organisation	Text	1..1
streetDescription	Street description of the organisation	Text	0..1
addressLine	Lines that allows specifying additional information concerning the exact location of the party	Text	0..5
postBox	Postbox of the organisation	Text	0..1
cityName	City name of the organisation	Text	1..1
region	Region of the organisation	Text	0..1
PurchaseCard			
This class contains information required by the seller to process purchase card payments			
cardNumber	The number of the card of the order point	Identifier	1..1
cardHolderName	The name of the holder of the card (with eventually the organisation name)	Text	1..1
expiryMonth	The month and year of the expiry date	Date	1..1
validFromMonth	The month and year from which the card is valid	Date	0..1
verificationValue	A digit number that suffixes the card number	Text	0..1
issueNumber	The issue number of the card	Text	0..1
CRI	Customer reference indicator (CRI), which is used to enable transmission of customer specific information with the card transaction.	Text	0..1
EFTaddress			
This class contains information on the bank account of the party, required to process funds transfer			
accountNumber	The account number of the accounts receivable or the invoice point	Identifier	1..1
branchID	The identifier of the branch the account is issued	Identifier	0..1
bankName	The name of the bank of the accounts receivable or the invoice point	Text	1..1
accountName	The name of the account of the accounts receivable or the invoice point	Text	0..1

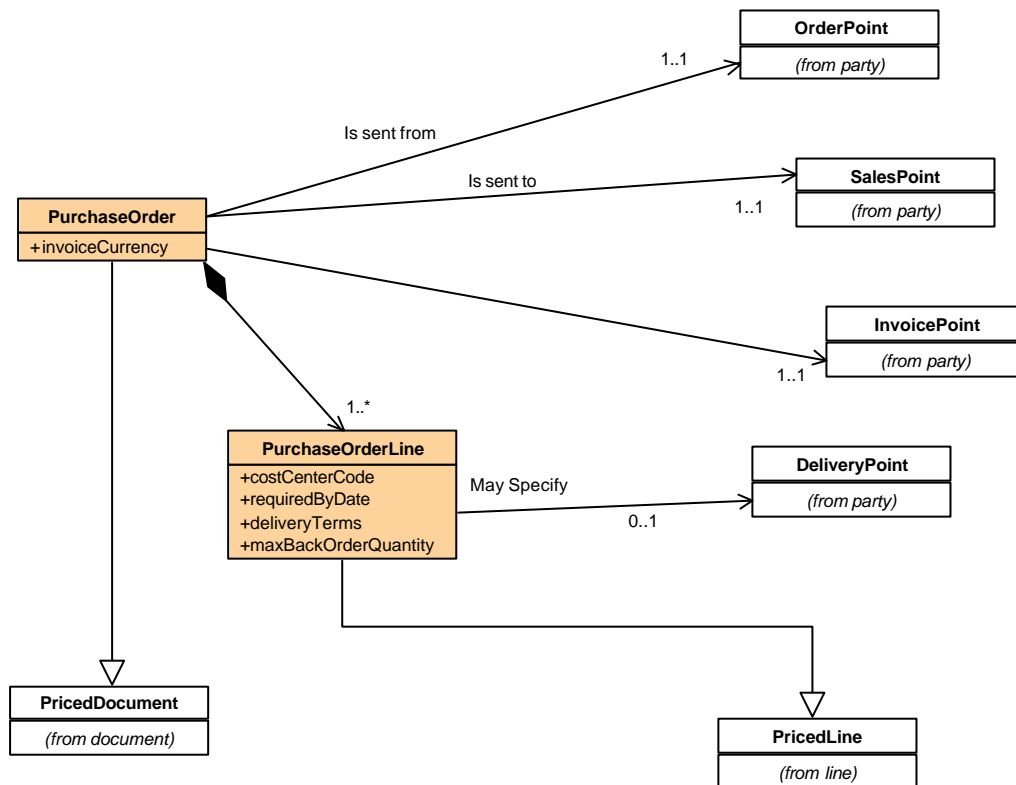
Attribute	Description	Type	Cardinality
SWIFTCode	The international bank transfer code of the bank of the accounts receivable or the invoice point	Identifier	0..1
OrderPoint			
Buyer party for issues related to purchase orders			
SalesPoint			
Supplier party responsible for purchasing issues prior to fulfilment of Purchase Order			
AccountsReceivable			
Supplier party for payment-related issues			
InvoicePoint			
Buyer party for payment-related issues			
DeliveryPoint			
Buyer party that receives goods/works/services and identifies variances in receipt			
DespatchPoint			
Supplier party responsible for the delivery of goods			
Carrier			
Third party responsible for the delivery of goods			
CustomerService			
Supplier party responsible for issues related to purchase order fulfilment (despatch, delivery and carriage)			
TaxRepresentative			
Supplier party responsible for keeping VAT records and accounts for taxable transactions in another Member State			

3.11 PURCHASE ORDER

The whole procurement cycle usually starts with a *PurchaseOrder* placed by the buyer.

Only one *Item* can be referenced in one *PurchaseOrderLine*. Nevertheless, there MAY be several lines corresponding to the same type of item, if different quantities of the same product have to be delivered at different point in time.

The **Purchase Order** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
PurchaseOrder			
This class inherits from PricedDocument and specifies the currency to be used for the invoice			
invoiceCurrency	ISO code of the currency to be used for the invoice	Code	0..1
PurchaseOrderLine			
This class inherits from Line and contains information relating to the delivery			
costCenterCode	Code used by the buyer for the type of goods being purchased.	Code	0..1
requiredByDate	Date by when the line items delivery is expected	DateTime	0..1
deliveryTerms	INCOTERM Code & qualifier (if any)	Code	0..1
maxBackOrderQuantity	The quantity the Supplier is allowed to back order	Quantity	0..1

3.12 PO RESPONSE

There SHOULD always be a *POResponse* to a *PurchaseOrder* although, depending on the trading agreement or practices between the buyer and the supplier, this MAY not be the case.

A *POResponse* MUST refer to one and only one *PurchaseOrder* whether it is paper or electronic.

A *POResponse* MUST repeat all the *PurchaseOrderLines* whether they are accepted, rejected or accepted in part. Thereby, the *POResponse* summarises the supplier's commitment.

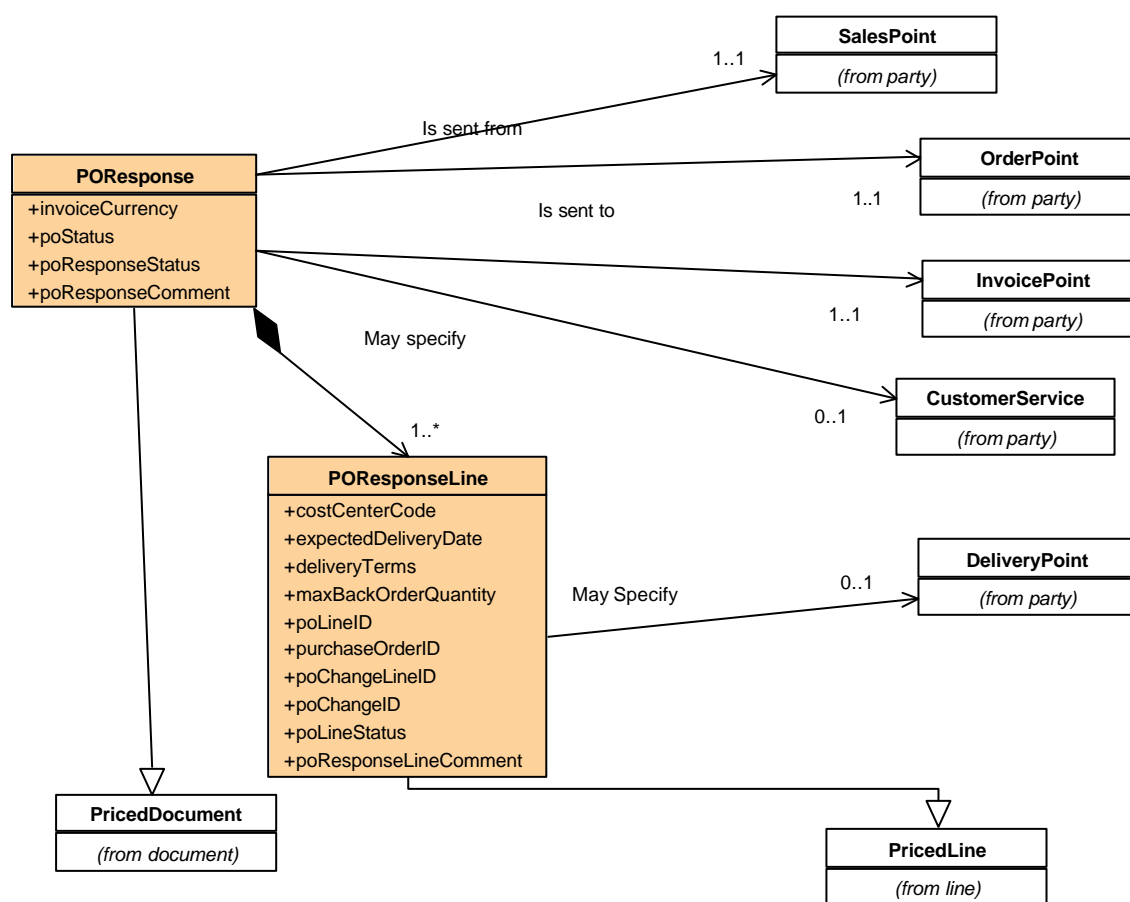
There MAY be several *POResponses* for the same *PurchaseOrder*.

All *POResponseLines* MUST explicitly reference a *PurchaseOrderLine*.

The supplier MAY respond to a *PurchaseOrderLine* with different *PO response lines* in particular when he plans several deliveries.

A *POResponse* contains a status (accepted, modified or rejected) for the whole of the document as well as a status for each line. Therefore, a "rejected" respectively "accepted" status at document level implicitly means that all the lines of the document are accordingly "rejected" respectively "accepted". A "modified" status at the document level implies that the status has to be specified for each line of the document mentioning "accepted", "rejected" or "modified".

The **Purchase Order Response** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
-----------	-------------	------	-------------

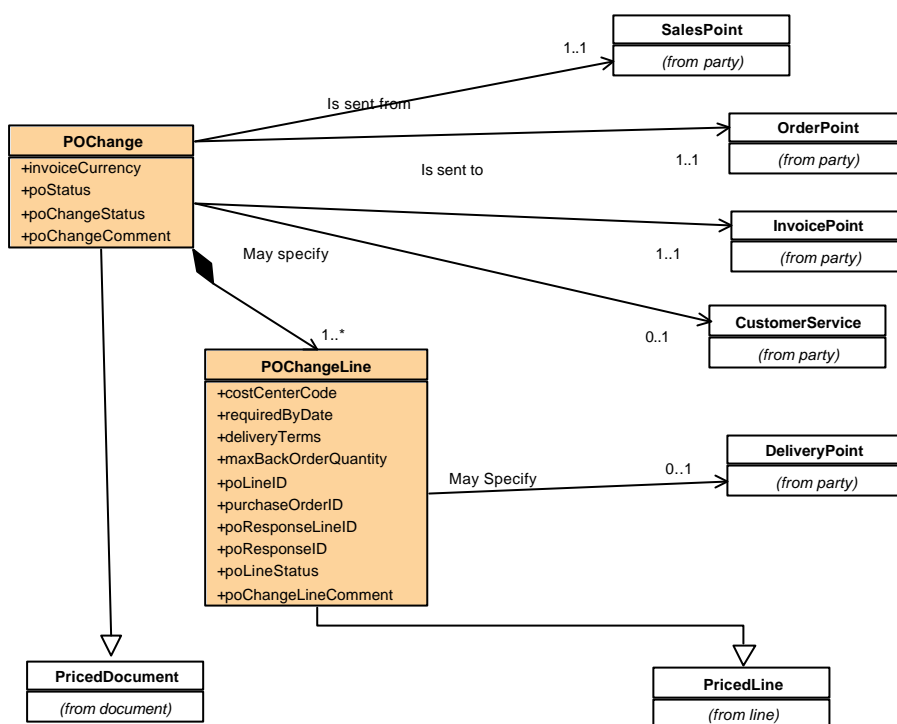
Attribute	Description	Type	Cardinality
POResponse			
This class inherits from PricedDocument and contains the same data as for the order as well as information about the Seller's answer to the order			
invoiceCurrency	ISO code of the currency to be used for the invoice	Code	0..1
poStatus	The purchase order status (Rejected, Modified, Accepted)	Code	1..1
poResponseStatus	The nature of modification in case of purchase order modification (Line addition, Line deleted, Price modification, Date, Quantity, Terms) Mandatory in case of poStatus="Modified"	Code	0..1
poResponseComment	Free text for the order response	Text	0..1
POResponseLine			
This class inherits from Line and contains information that allows commenting the answer as well as identifying to which order, order response and/or order change, the purchase order response line refers			
costCenterCode	Code used by the buyer for the type of goods being purchased.	Code	0..1
expectedDeliveryDate	Date by when the line items delivery is expected	DateTime	0..1
deliveryTerms	INCOTERM Code & qualifier (if any)	Code	0..1
maxBackOrderQuantity	The quantity of items the Supplier is allowed to back order	Quantity	0..1
poLineID	The identifier of the purchase order line that the current PO Response Line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order that the above PO Line refers to Mandatory if poLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line that the current PO Response Line refers to	Identifier	0..1
poChangeID	The identifier of the purchase order change that the above PO Change Line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
poLineStatus	The purchase order line status (Rejected, Modified, Accepted)	Code	1..1
poResponseLineComment	Free text for the response line	Text	0..1

3.13 PO CHANGE

A *POChange* is sent by the buyer to inform the seller that he wishes to modify or cancel the purchase order he has issued.

The structure of a *POChange* is similar to a *POResponse*. The only difference is that it MAY reference a previous *POResponse*.

The **Purchase Order Change** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
POChange			
This class inherits from PricedDocument and contains the same data as for the order or the order response as well as information about the Buyer's order modifications			
invoiceCurrency	ISO code of the currency to be used for the invoice	Code	0..1
poStatus	The purchase order status (Cancelled, Modified, Accepted)	Code	1..1
poChangeStatus	The nature of modification in case of purchase order modification (line modification, price modification, terms modification, ...)	Code	0..1
poChangeComment	Free text for the order change	Text	0..1
POChangeLine			
This class inherits from Line and contains information that allows commenting the modifications as well as identifying to which order and/or, order response, the purchase order change line refers			
costCenterCode	Code used by the buyer for the type of goods being purchased.	Code	0..1
requiredByDate	Date by when the line items delivery is expected	DateTime	0..1
deliveryTerms	INCOTERM Code & qualifier (if any)	Code	0..1
maxBackOrderQuantity	The quantity of items the Supplier is allowed to back order	Quantity	0..1
poLineID	The identifier of the purchase order line that the current PO Change Line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order that the above PO Line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line	Identifier	0..1

Attribute	Description	Type	Cardinality
	that the current PO Change Line refers to		
poResponseID	The identifier of the purchase order response that the above PO Response Line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poLineStatus	The purchase order line status (Cancelled, Modified, Accepted)	Code	1..1
poChangeLineComment	Free text for the purchase order change line	Text	0..1

3.14 DESPATCH ADVICE

A *DespatchAdvice* is sent by the seller to inform the buyer of that goods have been shipped to him or services fulfilled.

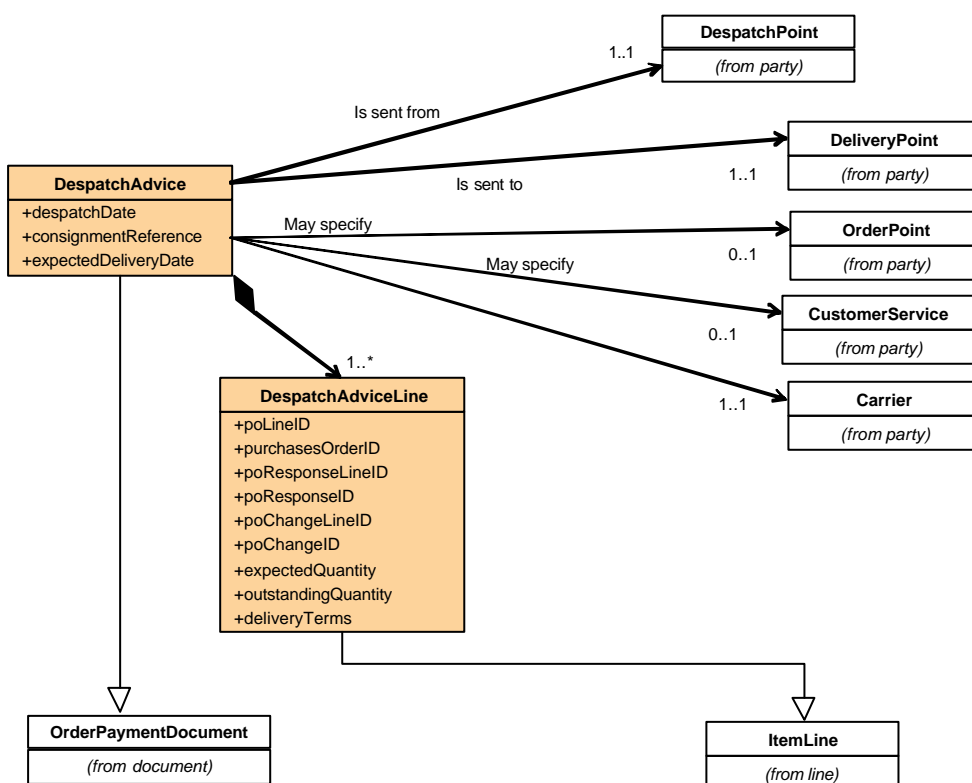
A *DespatchAdvice* corresponds to only one delivery (one or several items delivered at the same time).

A *DespatchAdvice* SHOULD reference one or more *POResponses* and/or *PurchaseOrders* and/or *POChanges* (i.e. goods or services ordered separately can be shipped and/or delivered at the same time).

There MAY, in case of partial delivery, be several *DespatchAdvices* for the same *PurchaseOrder*.

A *DespatchAdviceLine* SHOULD reference a *POResponseLine* and/or a *PurchaseOrderLine* and/or a *POChangeLine*.

The **Despatch Advice** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
DespatchAdvice			

Attribute	Description	Type	Cardinality
This class inherits from OrderPaymentDocument and contains general information about the despatch process			
despatchDate	The date when items are expected to be despatched (according to the related order)	DateTime	0..1
consignmentReference	The reference of the consignment related to this operation	Text	0..1
expectedDeliveryDate	The estimated time of delivery of the items	DateTime	0..1
DespatchAdviceLine			
This class inherits from Line and contains information that allows commenting the answer as well as identifying to which order, order response and/or order change, the dispatch advice line refers			
poLineID	The identifier of the purchase order line the current despatch advice line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order the above purchase order line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line the current despatch advice line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response the above order response line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line the current despatch advice line refers to	Identifier	0..1
poChangeID	The identifier of the purchase order change the above order change line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
expectedQuantity	The quantity of items that is expected in this delivery (according to the delivery terms)	Quantity	0..1
outstandingQuantity	The quantity of items still to be shipped (with regards to the delivery terms)	Quantity	0..1
deliveryTerms	The terms of delivery of the items	Text	0..1

3.15 RECEIPT ADVICE

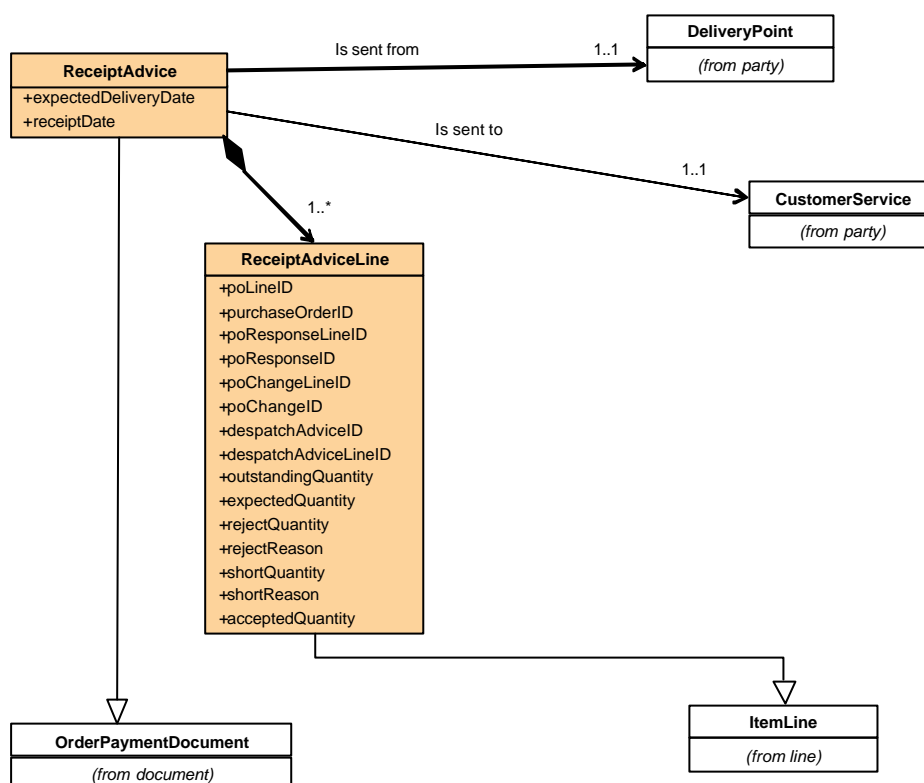
A Receipt Advice is sent by the buyer to the seller to acknowledge the total or partial reception of what has been ordered, indicating the variances related to the items, the delivery terms or any other differences.

A *ReceiptAdvice* SHOULD correspond to one and only one *DespatchAdvice*.

For one *DespatchAdvice*, there SHOULD be only one *ReceiptAdvice*.

A *ReceiptAdviceLine* SHOULD reference a *DespatchAdviceLine* as well as a *POResponseLine* and/or a *PurchaseOrderLine* and/or a *POChangeLine*.

The **Receipt Advice** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
ReceiptAdvice			
This class inherits from OrderPaymentDocument and contains the information relating to the date of items reception			
expectedDeliveryDate	The estimated time of delivery of the items (according to the despatch advice)	DateTime	0..1
receiptDate	The date when the goods were received or services completed by the Delivery Point	DateTime	1..1
ReceiptAdviceLine			
This class inherits from Line and contains quantitative and qualitative information relating to the items that have been delivered as well as data identifying to which order, order response, order change and/or dispatch advice, the receipt advice line refers			
poLineID	The identifier of the purchase order line the current receipt advice line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order the above purchase order line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line the current receipt advice line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response the above purchase order response line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line the current receipt advice line refers to	Identifier	0..1
poChangeID	The identifier of the purchase order change the above purchase order change line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1

Attribute	Description	Type	Cardinality
despatchAdviceLineID	The identifier of the despatch advice line the current receipt advice line refers to	Identifier	0..1
despatchAdviceID	The identifier of the despatch advice the above despatch advice refers to	Identifier	1..1
outstandingQuantity	The quantity of items still to be shipped (with regards to the delivery terms)	Quantity	0..1
rejectQuantity	The quantity of delivered items that are rejected by the Buyer	Quantity	0..1
rejectReason	Reason of the items rejection	Text	0..1
shortQuantity	The quantity of delivered items that are missing	Quantity	0..1
shortReason	Comment on the items that are missing	Text	0..1
acceptedQuantity	The quantity of delivered items that are accepted by the Buyer	Quantity	0..1

3.16 RECTIFICATION ADVICE

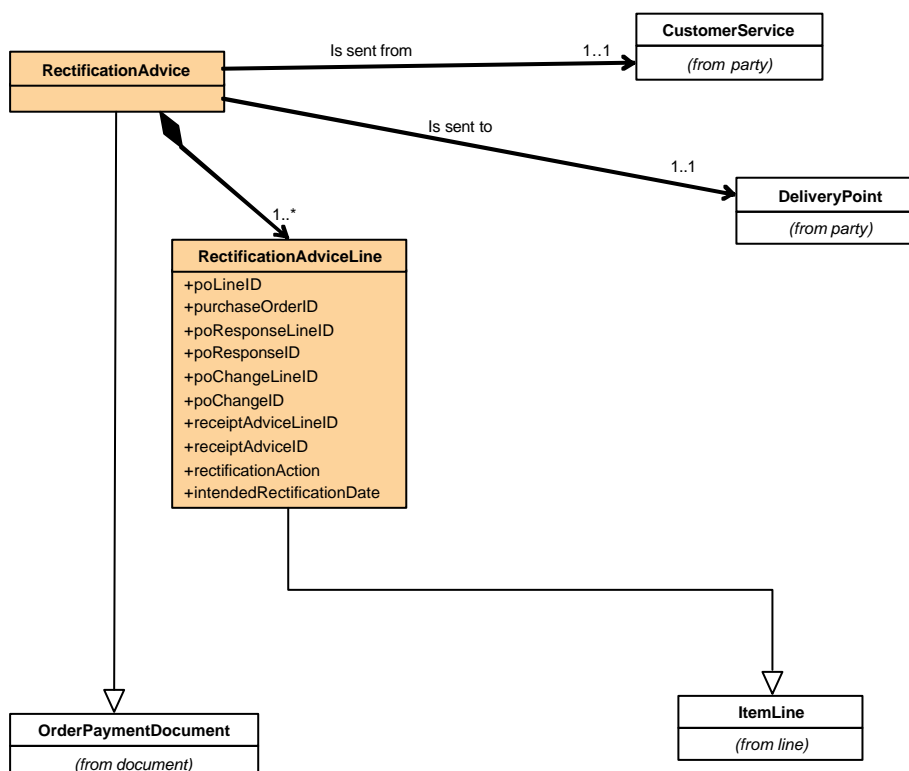
A Rectification Advice is sent by the seller to the buyer to inform him on the action to be taken according to the variances that have been observed on delivered items.

A *RectificationAdvice* SHOULD correspond to one and only one *ReceiptAdvice*.

For one *ReceiptAdvice*, there MAY be a single *RectificationAdvice*.

A *RectificationAdviceLine* SHOULD reference a (and only one) *ReceiptAdviceLine* as well as a *POResponseLine* and/or a *PurchaseOrderLine* and/or a *POChangeLine*.

The **Rectification Advice** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
RectificationAdvice			

Attribute	Description	Type	Cardinality
This class inherits from OrderPaymentDocument			
RectificationAdviceLine			
This class inherits from Line and contains information relating to the corrective actions that could be handle as well as data identifying to which order, order response, order change and/or receipt advice, the rectification advice line refers			
poLineID	The identifier of the purchase order line the current rectification advice line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order the above order line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line the current rectification advice line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response the above order response line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line the current rectification advice line refers to	Identifier	0..1
poChangeID	The identifier of the purchase order change the above order change line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
receiptAdviceLineID	The identifier of the receipt advice line the current rectification advice line refers to	Identifier	1..1
receiptAdviceID	The identifier of the receipt advice the above receipt advice line refers to	Identifier	1..1
rectificationAction	Free text that describes the corrective actions to be taken	Text	1..1
intendedRectificationDate	The intended date where the action could be taken	DateTime	0..1

3.17 INVOICE

Invoices are sent by the seller to the buyer for payment for delivered items or performed services.

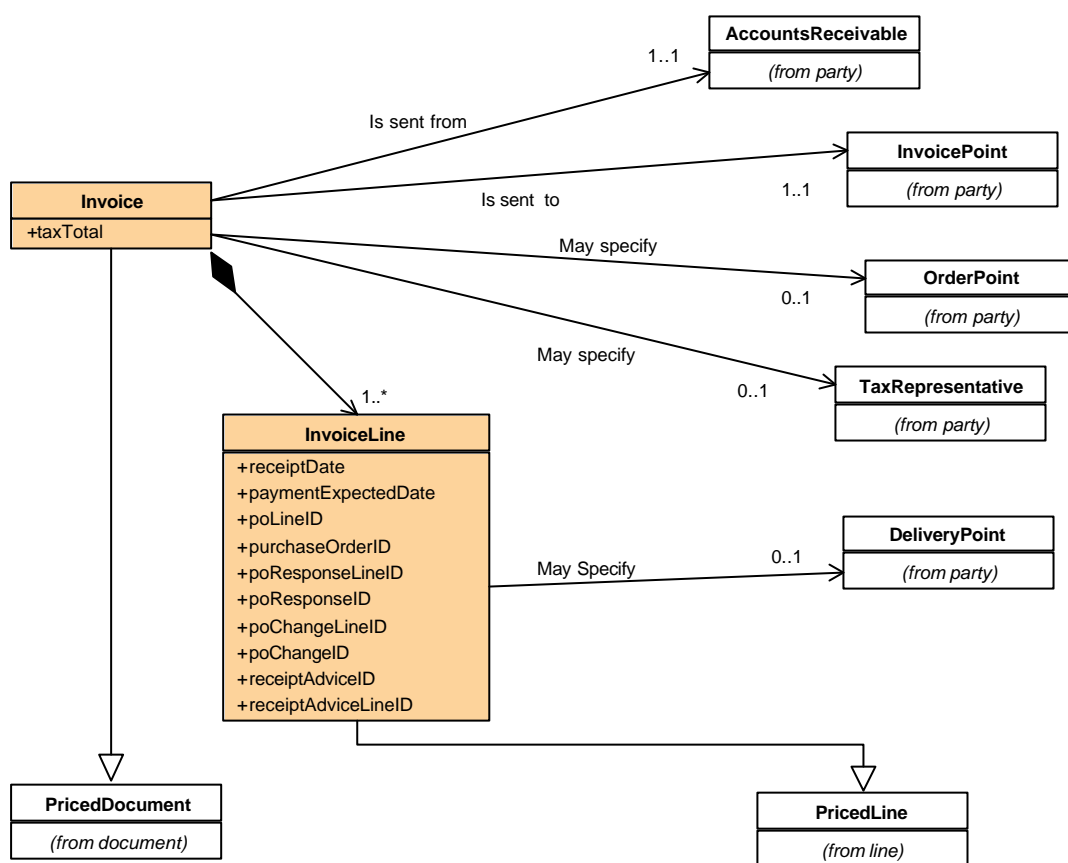
An *Invoice* MAY correspond to one or more *PurchaseOrders* in full or in part.

An *InvoiceLine* SHOULD correspond to only one *PurchaseOrderLine* as well as one *ReceiptAdviceLine*.

It is also recommended that a *PurchaseOrderLine* is invoiced only in one go, thereby corresponding to a unique *InvoiceLine* of a unique *Invoice*.

However, a *PurchaseOrderLine* MAY correspond to a repetitive provision of supplies or of a service and will subsequently lead to several invoices (placed by the seller on a regular basis).

The **Invoice** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
Invoice			
This class inherits from PricedDocument and contains information relating to the tax amount of the invoice			
taxTotal	The total amount of the tax related to the invoice	Amount	1..1
InvoiceLine			
This class inherits from Line and contains information that allows identifying to which order and order line the invoice line refers			
receiptDate	The date when the goods were received or services completed by the Delivery Point	DateTime	1..1
paymentExpectedDate	The expected date of payment or the effective date of payment if payment has already been made	DateTime	0..1
poLineID	The identifier of the purchase order line related to the invoice line	Identifier	0..1
purchaseOrderID	The identifier of the purchase order related to the invoice	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line that the current PO Change Line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response that the above PO Response Line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of another purchase order change line that the current PO Change Line refers to	Identifier	0..1

Attribute	Description	Type	Cardinality
poChangeID	The identifier of the purchase order change that the above PO Change Line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
receiptAdviceID	The identifier of the receipt advice line the current rectification advice line refers to	Identifier	0..1
receiptAdviceLineID	The identifier of the receipt advice the above receipt advice line refers to	Identifier	0..1

Note that the Invoice business document can also be used in a self-billing scenario.

3.18 INVOICE RESPONSE

An Invoice Response is sent by the buyer to inform the seller on discrepancies in the payment process that will probably require a credit note from the supplier. It may be associated with the purchase order as well as the receipt advice and the rectification advice, if relevant.

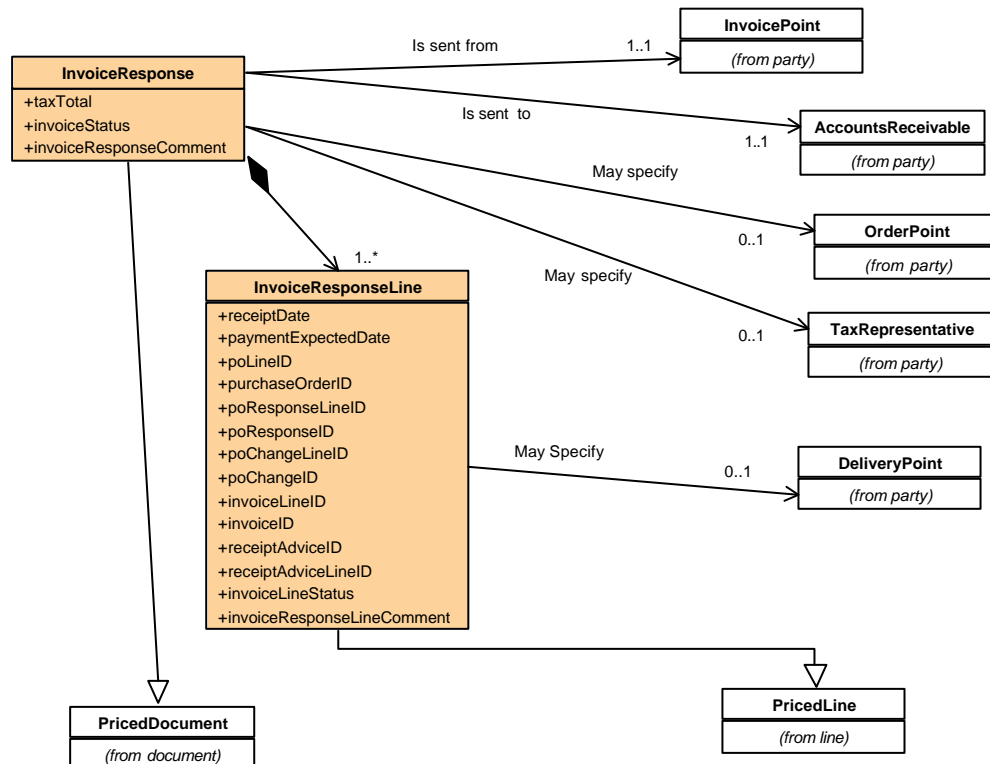
An *InvoiceResponse* MUST refer to one and only one *Invoice* whether it is paper or electronic.

An *InvoiceResponse* MUST repeat all the *InvoiceLines* whether they are accepted, rejected or accepted in part. Thereby, the *InvoiceResponse* summarises the buyer's commitment.

There MAY be several *InvoiceResponses* for the same *Invoice* although this SHOULD not be the case.

Each *InvoiceResponseLine* MUST reference an *InvoiceLine*.

The **Invoice Response** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
InvoiceResponse			
This class inherits from PricedDocument and contains the same data as for the invoice as well as			

Attribute	Description	Type	Cardinality
information about the buyer's answer to the invoice			
taxTotal	The total amount of the tax related to the invoice	Amount	0..1
invoiceStatus	The invoice status (Cancelled, Modified, Accepted)	Code	1..1
invoiceResponseComment	Free text for the response	Text	1..1
InvoiceResponseLine			
This class inherits from Line and contains information that allows commenting the answer as well as identifying to which order, order response, order change and/or invoice, the invoice response line refers			
receiptDate	The date when the goods were received or services completed by the Delivery Point	DateTime	1..1
paymentExpectedDate	The expected date of payment (according to the related order's payment schedule) or the effective date of payment (if payment has already been made)	DateTime	0..1
poLineID	The identifier of the purchase order line that the current invoice Response Line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order that the above purchase order line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line that the current invoice Response Line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response that the above order response line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line that the current invoice Response Line refers to	Identifier	0..1
poChangeID	The identifier of the purchase order change that the above order change line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
invoiceLineID	The identifier of the invoice line that the current invoice Response Line refers to	Identifier	0..1
invoiceID	The identifier of the invoice that the above invoice line refers to	Identifier	0..1
receiptAdviceID	The identifier of the receipt advice line the current rectification advice line refers to	Identifier	0..1
receiptAdviceLineID	The identifier of the receipt advice the above receipt advice line refers to	Identifier	0..1
invoiceLineStatus	The invoice line status (Cancelled, Modified, Accepted)	Code	1..1
invoiceResponseLineComment	Free text for the invoice response line	Text	0..1

3.19 CREDIT NOTE

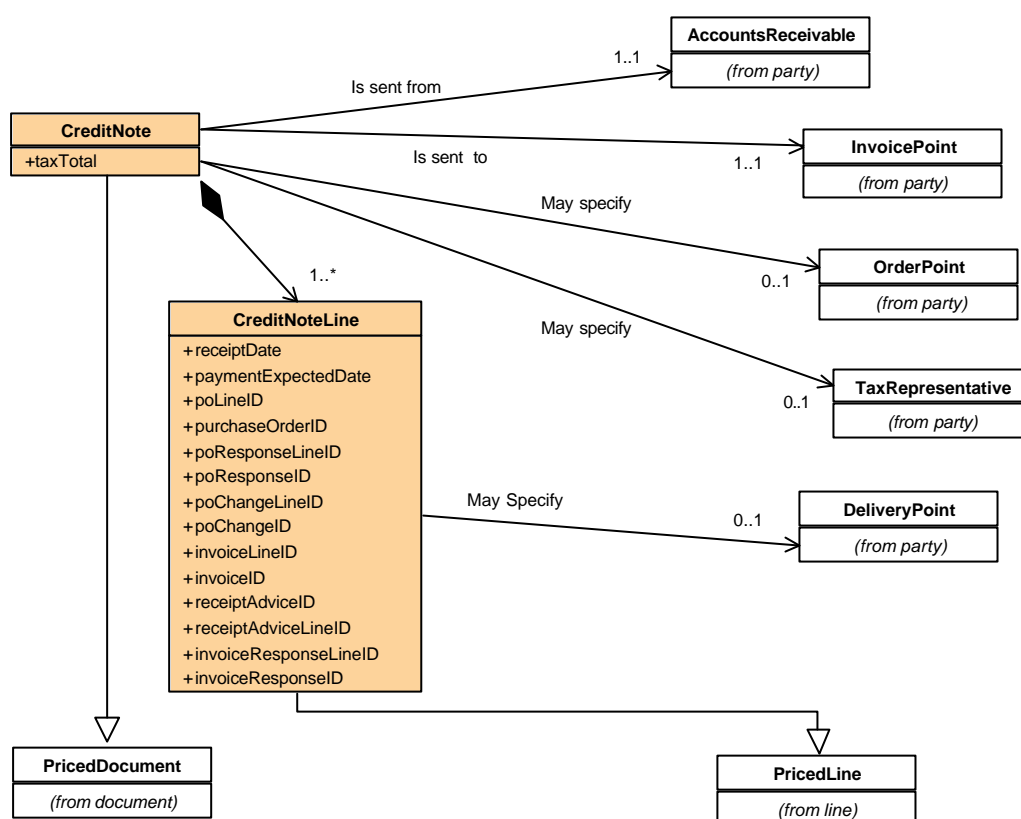
A Credit note is sent by the seller to confirm to the buyer that the buyer owes money to the supplier. It follows an invoice response.

A *CreditNote* can be seen as a “negative” invoice. Therefore, it has a similar structure to an *Invoice*.

A *CreditNote* MUST make reference to at least one *Invoice* but MAY reference several ones.

A *CreditNoteLine* MUST reference one and only one *InvoiceLine*.

The **Credit Note** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
CreditNote			
This class inherits from PricedDocument and contains information about the amount of the credit note			
taxTotal	The total amount of the credit note document	Amount	1..1
CreditNoteLine			
Line of a CreditNote			
receiptDate	The date when the goods were received or services completed by the Delivery Point	DateTime	1..1
paymentExpectedDate	The expected date of payment (according to the related order's payment schedule) or the effective date of payment (if payment has already been made)	DateTime	0..1
poLineID	The identifier of the purchase order line that the current invoice Response Line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order that the above purchase order line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line that the current invoice Response Line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response that the above order response line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line that the current invoice Response Line refers to	Identifier	0..1

Attribute	Description	Type	Cardinality
poChangeID	The identifier of the purchase order change that the above order change line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
invoiceLineID	The identifier of the invoice line that the current invoice Response Line refers to	Identifier	0..1
invoiceID	The identifier of the invoice that the above invoice line refers to	Identifier	0..1
receiptAdviceID	The identifier of the receipt advice line the current rectification advice line refers to	Identifier	0..1
receiptAdviceLineID	The identifier of the receipt advice the above receipt advice line refers to	Identifier	0..1
invoiceResponseLineID	The identifier of the invoice response line the current rectification advice line refers to	Code	1..1
invoiceResponseID	The identifier of the invoice response the above receipt advice line refers to	Text	0..1

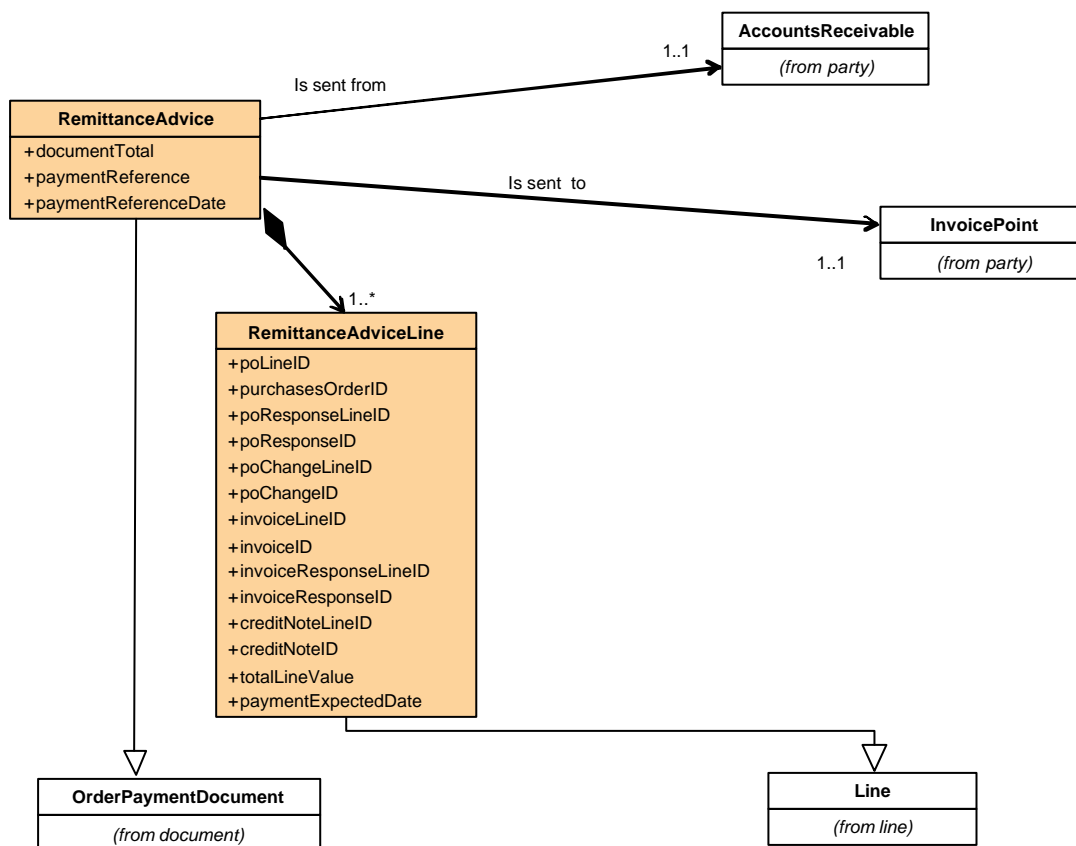
3.20 REMITTANCE ADVICE

A *RemittanceAdvice* is sent by the buyer to notify the seller that payment has been made.

A *RemittanceAdvice* MAY correspond to one or more *Invoice* or *InvoiceResponse*.

Each *RemittanceAdviceLine* MUST refer to one single *InvoiceLine* or *InvoiceResponseLine*.

The **Remittance Advice** UML class diagram is illustrated below.



Attribute	Description	Type	Cardinality
RemittanceAdvice			
This class inherits from OrderPaymentDocument and contains general information concerning the payment			
documentTotal	The total amount of the remittance document	Amount	1..1
paymentReference	The reference of the payment the remittance refers to	Text	0..1
paymentReferenceDate	The reference date of the payment the remittance refers to	DateTime	0..1
RemittanceLine			
This class inherits from Line and contains information relating to the line amount as well as data identifying to which order, order response, order change, invoice, invoice response and/or credit note, the remittance advice line refers			
poLineID	The identifier of the purchase order line the current remittance line refers to	Identifier	0..1
purchaseOrderID	The identifier of the purchase order the above purchase order line refers to Mandatory if poLineID is not empty	Identifier	0..1
poResponseLineID	The identifier of the purchase order response line the current remittance line refers to	Identifier	0..1
poResponseID	The identifier of the purchase order response the above purchases order response line refers to Mandatory if poResponseLineID is not empty	Identifier	0..1
poChangeLineID	The identifier of the purchase order change line the current remittance line refers to	Identifier	0..1
poChangeID	The identifier of the purchase order change the above purchase order change line refers to Mandatory if poChangeLineID is not empty	Identifier	0..1
invoiceLineID	The identifier of the invoice line the current remittance line refers to	Identifier	0..1
invoiceID	The identifier of the invoice the above invoice line refers to Mandatory if invoiceLineID is not empty	Identifier	0..1
invoiceResponseLineID	The identifier of the invoice response line the current remittance line refers to	Identifier	0..1
invoiceResponseID	The identifier of the invoice response the above invoice response line refers to Mandatory if invoiceResponseLineID is not empty	Identifier	0..1
creditNoteLineID	The identifier of the credit note line the current remittance line refers to	Identifier	0..1
creditNoteID	The identifier of the credit note the above credit note line refers to Mandatory if creditNoteLineID is not empty	Identifier	0..1
totalLineValue	The total amount of the remittance line	Amount	0..1
paymentExpectedDate	The expected date of the payment (according to the related order's payment schedule) the remittance refers to	DateTime	0..1

4 EXTENDING THE IDA E-PROCUREMENT MODEL

1. The easiest way to add extra-information is to use the free-text attributes provided by the IDA model at different level:

- At *Document* level (*Document.documentNote*) when one needs to specify information related to the whole *Document*;
- At *Line* level (*Line.lineNote*) when one needs to specify information related to a particular *Line*;
- Related to a *Party* (*Party.partyNote*) when extra-information concerning one of the parties involved is required.

This however presents limitations:

- Those attributes are free-text that can contain varied information; thus, automatic processing of their content can barely be envisaged;
- The content of those attributes is unstructured and cannot easily hold structured data.

2. Should you require extra-data that you want to process automatically and/or that can hold structured data, you may extend of any of the *complexType*s defined in our model by deriving them⁹.

⁹ An example is given at the following address <http://www.w3.org/TR/xmlschema-0/#DerivExt>.
IDA e-procurement protocol

5 GAP WITH UBL

The UBL model is globally more complete and more complex than the IDA one although very similar in scope.

The main differences are listed below:

- In the IDA model, business documents contain references to all the relevant other business documents or lines they are associated with. For instance, (at both document and line level) a receipt advice may refer to a purchase order or a purchase order response or a purchase order change, and the related dispatch advice.
- Some information exists only in the UBL model
 - Acknowledgement response attribute
 - Shipping information
 - Copy indicator attribute
 - GUID attribute
 - Language attribute
 - Base price class
 - Contract message
 - Allowance charge
- Some information exists only in the IDA model
 - costCentre attribute
 - ItemInstance class
 - Charges and discounts information
 - Invoice Response message
 - Credit Note message
 - Remittance advice message
 - Rectification advice message
- UBL model describes in more details packaging, transport and shipping information
- Currency can be specified for any business documents, at the document level, in the IDA model
- The line Item class is more complex in the UBL model
- Tax and Address information is more accurate in the UBL model
- Order cancellation is covered in the same message than Order Change or Order Response in the IDA model
- There are more details describing the items in the UBL model
- Payment, terms and conditions information is structured in different attributes in the IDA model
- Detailed information on the exchange currency is managed in the IDA model.

The following table details the mapping between the two models for the *PurchaseOrder* and *Invoice* business document and their related information components.

UBL	IDA
ORDER	
Order	PurchaseOrder
BuyersID	Document.documentID
SellersID	Not managed
CopyIndicator	
GUID	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
IssueDate	Document.documentDate
Note	Document.documentNote
AcknowledgementResponseCode	Part of the <i>PO Response</i> document.
TransactionCurrencyCode	Part of the Amount datatype.
PricingCurrencyCode	Not managed as a separate attribute but an <i>currency</i> attribute is attached to all Amount attribute.
EarliestDate	Not managed
CancelledByDate	Similar to PurchaseOrder.requiredByDate
ValidityDurationMeasure	Not managed

UBL	IDA
TaxTotalAmount	TaxTypeTotal.taxTypeTotal
LineExtensionTotalAmount	PricedLine.netLineValue
TotalPackagesQuantity	
GrossWeightMeasure	
NetWeightMeasure	
NetNetWeightMeasure	
GrossVolumeMeasure	
NetVolumeMeasure	
LineItemCountQuantity	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC12)
OrderLine	PurchaseOrderLine
Not managed	deliveryTerms
	costCenterCode
SubstitutionStatusCode	Not managed
Note	Line.lineNote
LineItem	Item & Line
BuyersID	Not managed
SellersID	Item.sellerItemID
LineStatusCode	Not managed
Quantity	QuantifiedLine.lineQuantity
LineExtensionAmount	PricedLine.netLineValue
TaxTotalAmount	PricedLine->LineTax.lineTax
MinimumQuantity	Not managed
MaximumQuantity	
MaximumBackorderQuantity	PurchaseOrderLine.maxBackOrderQuantity
MinimumBackOrderQuantity	Not managed
Note	Item.itemDescription
INVOICE	
Invoice	Document & Invoice
ID	Document.documentID
CopyIndicator	Not managed
GUID	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
IssueDate	Document.documentDate
	Not managed. Note that such information might not be necessary since a CreditNote is a separate business document in the IDA model.
InvoiceTypeCode	
Note	Document.documentNote
TaxPointDate	Managed differently (at line level): InvoiceLine.receiptDate
InvoiceCurrencyCode	PurchaseOrder.invoiceCurrency
TaxCurrencyCode	PricedDocument->TaxTypeTotal.taxTypeTotal (currency attribute)
PricingCurrencyCode	Managed at line level: Line.netLineValue (currency attribute)
LineItemCountQuantity	Not managed.
InvoiceLine	InvoiceLine
ID	Line.lineID
LineStatusCode	Not managed

UBL	IDA
InvoiceQuantity	Line.lineQuantity
LineExtensionAmount	PricedLine.netLineTotal
Note	Line.lineNote
Managed differently (at document Level). Invoice.TaxPointDate)	receiptDate
Not managed	paymentExpectedDate
	poLineID
	purchaseOrderID
	poResponseLineID
	poResponseID
	poChangeLineID
	poChangeID
	receiptAdviceLineID
	receiptAdviceID
PARTY	
Party	Party
	NOT AN "EMPTY" CLASS
SellerParty	Not managed
BuyerAssignedAccountID	
SellerAssignedAccountID	
AdditionalAccountID	
BuyerParty	Not managed
BuyerAssignedAccountID	
SellerAssignedAccountID	
AdditionalAccountID	
Contact	ContactInformation
ID	Not managed
Name	contactName
Telephone	directLine
Telefax	fax
ElectronicMail	email
Not managed	jobTitle
	switchboard
	mobilePhone
Communication	Not managed
ChannelCode	
Value	
PartyIdentification	Managed differently. In the IDA model parties can be identified by Party.gln, Party.dunsNumber or Party.taxNumber)
ID	
PartyName	Party
Name	organisationName
Not managed	partyNote

UBL	IDA
	dunsNumber
	Gln
	taxNumber
	registrationNumber
	registrationCountryCode
Language	All Text attributes may have a language attached to it.
ID	
Name	
LocaleCode	
Not managed	OrderPoint
Not managed	SalesPoint
Not managed	InvoicePoint
Not managed	DespatchPoint
Not managed	DeliveryPoint
Not managed	Carrier
Not managed	CustomerService
Not managed	AccountsReceivable
BasePrice	Not managed.
PriceAmount	
BasesQuantity	
MinimumQuantity	
MaximumQuantity	
MinimumAmount	
MaximumAmount	
TAX	
TaxTotal	Invoice & Line
TotalTaxAmount	Invoice.taxTotal & LineTax.lineTax
TaxSubTotal	Document, Line, Invoice
TaxableAmount	PricedDocument.netDocumentTotal & PricedLine.netLineValue
TaxAmount	Invoice.taxTotal & LineTax.lineTax
TaxCategory	StandardTaxData
ID	Not managed
RatePercentNumeric	taxPerCent
Not managed	taxBand
PartyTaxScheme	Party

UBL	IDA
RegistrationName	Not managed. For the VAT, an equivalent would be the Party.taxNumber.
Company	organisationName
TaxLevelCode	
ExemptionReasonCode	Not managed
TaxScheme	StandardTaxData
ID	Not managed.
TaxTypeCode	taxType
CurrencyCode	Not managed.
CONTRACT	
ID	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC15)
IssueDate	
ContractTypeCode	
Period	
StartDateTime	
EndDateTime	
DurationMeasure	
DescriptionCode	
ADDRESS	
Address	Address
	Not managed as a specific attribute but the addressLine attribute can be used for all extra-information on an address.
ID	
postbox	postBox
Floor	Not managed as a specific attribute but the addressLine attribute can be used for all extra-information on an address.
Room	
StreetName	streetDescription
AdditionalStreetName	Not managed as a specific attribute but the addressLine attribute can be used for all extra-information on an address.
BuildingName	
BuildingNumber	
InhouseMail	
Department	ContactInformation.department
CityName	cityName
PostalZone	Postcode
CountrySubentity	Not managed as a specific attribute but the addressLine attribute can be used for all extra-information on an address.
CountrySubentityCode	
Region	Region
District	Not managed as a specific attribute but the addressLine attribute can be used for all extra-information on an address.
TimezoneOffset	
Country	Address
IdentificationCode	countryCode
Name	Not managed
AddressLine	Address
Line	addressLine

UBL	IDA
LocationCoordinate (latitude, longitude)	Not managed
Country	Address
IdentificationCode	code of all attribute of type <i>Country</i> .
Name	Value of all attribute of type <i>Country</i> .
CommodityClassification	Item
NatureCode	Not managed.
CargoTypeCode	
CommodityCode	
	commodityClass
SalesConditions	Not managed.
ID	
ActionCode	
Description	
PhysicalAttribute	Aspect
AttributeID	Not managed
PositionCode	
DescriptionID	
Description	aspectName & aspectValue & aspectValueUnit
Dimension	Not managed
AttributeID	
Measure	
Description	
MinimumMeasure	
MaximumMeasure	
Not managed	ItemInstance
	instanceSerialNumber
	instanceBatchNumber
	instanceSupplierReference
	instanceRegistrationNumber
	instanceManufactureDate
	instanceRegistrationDate
	instanceExpiryDate
DOCUMENT REFERENCE	
LineReference	Line
LineID	lineID
lineStatusCode	Not managed
Not managed	lineNote
Not managed	PricedLine
	totalLineValue
	chargesLineValue
	discountsLineValue

UBL	IDA
Not managed	ItemLine
DocumentReference	Document
ID	documentID
CopyIndicator	Not managed
IssueDate	documentDate
GUID	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
	OrderPaymentDocument
	contractReference
	paymentMethod
	settlementTerms
	termsConditions
	paymentSchedule
	digitalSignature
OrderLineReference	Not managed as a separate class but with a list of attributes
BuyersLineID	
SellersLineID	
LineStatusCode	
OrderReference	Not managed as a separate class but with a list of attributes
BuyersID	
SellersID	
CopyIndicator	
IssueDate	
GUID	
DocumentReference	PricedDocument
	documentTotal
Not managed	chargesTotal
	discountsTotal
	Attachment
	attachment
	filename
	description
	size
	MIMEType
	ExchangeCurrency
	targetCurrencyCode
	baseCurrencyCode
	targetToBaseRate
	exchangeRateDate
	exchangeRateSource
Branch	EFTaddress
ID	branchID

UBL	IDA
Name	bankName
Country	Address
IdentificationCode	countryCode
Name	Not managed
FinancialInstitution	Not managed
ID	
Name	

6 GAP WITH EHADEL

The eHandel model is globally more limited in scope (covers mainly Ordering and invoicing) and is based on the assumption that it is related to e-commerce conducted through the eHandel marketplace.

The main differences are listed below:

- In the IDA model, business documents contain references to all the relevant other business documents or lines they are associated with. For instance, (at both document and line level) a receipt advice may refer to a purchase order or a purchase order response or a purchase order change, and the related dispatch advice.
- Some information exists only in the eHandel model
 - Language attribute
 - OffCatalogueFlag attribute
 - Account information
 - PriceCheckRequest message
 - PriceCheckResult message
 - AvailabilityCheckRequest message
 - AvailabilityCheckResult message
- Some information exists only in the IDA model
 - Document, Line, Item, Party reusable packages
 - Charges and discounts information
 - Item instance information
 - Item aspect information
 - Purchase card information
 - Invoice response message
 - Despatch advice message
 - Rectification advice message
 - Credit note message
 - Remittance advice
- Detailed information on the exchange currency is managed in the IDA model
- Party description, Order contact as well as EFT address are less exhaustive in the eHandel model
- Currency can be specified for any business documents, at the document level, in the IDA model
- Totals have to be expressed also in the tax currency in the eHandel model

The following table details the mapping between the two models for the *PurchaseOrder* and *Invoice* business document and their related information components.

eHandel	IDA
Order	PurchaseOrder
buyerOrderNumber	Document.documentID
orderIssueDate	Document.documentDate
costCenter	PurchaseOrderLine.costCenterCode
purpose	Not managed as a specific attribute
headerCurrency	Part of each <i>Amount</i> attribute
language	Part of each <i>Text</i> attribute.
requestedShipByDate	Not managed
requestedDeliverByDate	PurchaseOrderLine.expectedDeliveryDate
requisitionDate	Not managed
termsOfDelivery	PurchaseOrderLine.deliveryTerms
transportTerms	
shipmentMethodOfPayment	
transportDescription	
paymentTerm	OrderPaymentDocument.settlementTerms
netDaysDue	OrderPaymentDocument.paymentSchedule
netDueDate	
netDateTimeRef	
paymentMean	OrderPaymentDocument.paymentMethod

eHandel	IDA
headerNote	Document.documentNote
headerAttachment	Attachment.attachment
totalAmount	PricedDocument.documentTotal
Not managed	OrderPaymentDocument
	contractReference
	termsConditions
Not managed	Document
	digitalSignature
Not managed	StandardTaxData
	taxType
	taxBand
Not managed	PricedDocument
	chargesTotal
	discountsTotal
Not managed	ExchangeCurrency
	targetCurrencyCode
	baseCurrencyCode
	targetToBaseRate
	exchangeRateDate
	exchangeRateSource
Not managed	OrderPoint
Not managed	SalesPoint
Not managed	InvoicePoint
Not managed	DespatchPoint
Not managed	DeliveryPoint
Not managed	Carrier
Not managed	CustomerService
Not managed	AccountsReceivable
OrderItem	Line & Item
lineItemNumber	Line.lineID
productDescription	Item.itemDescription
commodityCode	Item.commodityClass
totalQuantity	QuantifiedLine.lineQuantity
maxBackOrderQuantity	PurchaseOrderLine.maxBackOrderQuantity
offCatalogueFlag	Not managed.
unitPrice	PricedItem.unitPrice
monetaryAmount	PricedLine.netLineValue
requestedDeliveryDate	PurchaseOrderLine.expectedDeliveryDate

eHandel	IDA
itemNote	Item.itemDescription
itemAttachment	Attachment.attachment
Not managed	Line
	lineNote
Not managed	PricedLine
	totalLineValue
	chargesLineValue
	discountsLineValue
Not managed	ItemLine
	ItemInstance
Not managed	instanceSerialNumber
	instanceBatchNumber
	instanceSupplierReference
	instanceRegistrationNumber
	instanceManufactureDate
	instanceRegistrationDate
	instanceExpiryDate
Not managed	Item
	itemName
	unitOfMeasure
	Gtin
Not managed	Aspect
	aspectName
	aspectValue
	aspectValueUnit
Product	Item
supplierPartID	Item.sellerItemID
manufacturerPartID	
buyerPartID	Not managed
TaxInformation	Party
taxRegistrationID	Party.taxNumber
registeredName	
resgisteredOffice	Not managed
companyResgistration	Party.registrationNumber
Not managed	Party
	partyNote
	dunsNumber
	Gln
	registrationCountryCode
SellerBankInformation	EFTaddress
financialInstituteName	EFTaddress.bankName
bankAccount	EFTaddress.accountNumber

eHandel	IDA
Not managed	accountNumber
	branched
	SWIFTCode
Not managed	PurchaseCard
	cardNumber
	cardHolderName
	expiryMonth
	validFromMonth
	verificationValue
	issueNumber
	CRI
OrderContact	ContactInformation
contactName	contactName
telephoneNumber	directLine
faxNumber	Fax
emailAddress	Email
Not managed	jobTitle
	mobilePhone
	switchboard
	department
InvoiceVatSummary	StandardTaxTotal, TaxTypeTotal, PricedDocument
vatRate	StandardTaxTotal.taxPercent
vatAmountTotal	TaxTypeTotal.taxTypeTotal
taxableAmountTotal	PricedDocument.netDocumentTotal
vatAmountTotalInTaxCurrency	Not managed
taxableAmountTotalInTaxCurrency	
Account	Not managed
accountCode	
supplierAccountCode	
description	
supplierApprove	
allowBackorder	
allowPartialShipment	
preferredSupplier	
Party	Party & Address
identifier	Not managed
name1	Only one name: Party.organisationName
name2	
name3	
houseNumber	Not managed as a specific attribute but the <i>addressLine</i> attribute can be used for all extra-information on an address.
street	Address.streetDescription
building	Not managed as a specific attribute but the <i>addressLine</i> attribute can be used for all extra-information on an address.
roomNumber	
inhouseMail	
postalCode	Address.postCode
city	Address.cityName

eHandel	IDA
region	Address.region
country	Address.countryCode
Not managed	Address
	addressLine
	postBox
Invoice	Invoice
invoiceNumber	Document.documentID
invoiceDate	Document.documentDate
paymentReference	RemittanceAdvice.paymentReference
invoicePurpose	Not managed
invoiceType	Not managed. Note that such information might not be necessary since a CreditNote is a separate business document in the IDA model.
invoiceCurrency	PurchaseOrder.invoiceCurrency
paymentCurrency	Not managed
taxAccountingCurrency	PricedDocument->TaxTypeTotal.taxTypeTotal (currency attribute)
language	managed with the Language attribute on all Text value.
medium	
dueDate	Not managed.
paymentTerm	OrderPaymentDocument.settlementTerms
paymentMean	OrderPaymentDocument.paymentMethod
headerNote	Document.DocumentNote
netAmount	PricedDocument.netDocumentTotal
vatAmount	Invoice.taxTotal
grossAmount	PricedDocument.DocumentTotal
InvoiceItem	InvoiceLine
lineItemNumber	Line.lineID
productDescription	Item.itemDescription
commodityCode	Item.commodityClass
totalQuantity	Line.lineQuantity
offCatalogueFlag	Not managed
purchaseOrderNumber	purchaseOrderID
purchaseOrderDate	Not managed
purchaseOrderLineNumber	poLineID
supplierOrderNumber	Not managed
deliveryNoteNumber	receiptAdviceID
relatedInvoiceNumber	Invoice.invoiceID
unitPrice	PricedItem.unitPrice
vatRate	StandardTaxData.taxPerCent
vatAmount	LineTax.lineTax
vatAmountInTaxCurrency	Not managed
taxableAmount	PricedLine.netLineTotal
taxableAmountInTaxCurrency	Not managed
itemAmount	PricedLine.netLineTotal
itemAmountInTaxCurrency	Not managed but can be specified at document level with the PricedDocument->ExchangeCurrency class.
itemNote	Item.itemDescription

7 GAP WITH OGC

The OGC model is very close to the IDA one in terms of scope and design.

One of the major differences is that the IDA model does not cover sourcing and that it is significantly simpler (globally fewer classes/attributes and simpler design).

The main differences are listed below:

- in the IDA model, business documents contain references to all the relevant other business documents or lines they are associated with. For instance, (at both document and line level) a credit note may refer to a purchase order or a purchase order response or a purchase order change, the related receipt advice, the related invoice or invoice response.
- Some information exists only in the OGC model:
 - Document metadata class (apart from the digital signature attribute)
 - Shipping Information class
 - Priced item validity class
 - RFQ message
 - Catalogue update message: this is not part of our model for now; nevertheless, the exchange of catalogues have been identified as one of the ways to significantly streamline e-procurement processes and will be studied in a second time;
 - Draft supply requisition message
 - Statement message
 - Debit note message
- Some information exists only in the IDA model:
 - MaxBackOrderQuantity attribute
 - Purchase order change class
- In the OGC model, at the line level, only charges and discounts totals are managed (no detail is available)
- Address information is less complex in the IDA model
- Receipt Advice information is more structured within the IDA model
- The Credit Note class is more complex in the IDA model since it contains the same information as for the Invoice class
- In the IDA model, responses contain a specific attribute (different from the ones available in the Document and Line packages) allowing describing the comment and the status of the response
- The OGC model manages several identifiers for business documents.

The following table details the mapping between the two models for the *PurchaseOrder* and *Invoice* business document and their related information components.

OGC	IDA
DOCUMENT PACKAGE	
DocumentID	Document
documentID	documentId
documentDate	documentDate
documentStatus	Only specific to business document type (PO Response, PO Change, etc.)
docUuid	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
Digital Signature	Document
digitalSignature	digitalSignature
DocumentAttachment	Attachment
attachment	attachment
	filename
	description
	size
Not managed	MIMEType

OGC	IDA
Document	Document
documentNote	documentNote
contractRef	OrderPaymentDocument.contractReference
paymentMethod	OrderPaymentDocument.paymentMethod
settlementTerms	OrderPaymentDocument.settlementTerms
termsConditions	OrderPaymentDocument.termsConditions
paymentSchedule	OrderPaymentDocument.paymentSchedule
ReferencedDocument	Not managed as a specific class but by specific attributes (referring to a document id).
PricedDocument	PricedDocument
DocumentMetaData	Not managed
testStatus	
senderSWManufacturer	
senderSWProduct	
senderSWVersion	
schemaVersion	
styleSheetReference	
senderSystemID	
sendersLogo	
LINE PACKAGE	
LineID	Line
lineNumber	lineID
lineUuid	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
LineAttachment	Document package.Attachment
attachment	attachment
Line	Line
lineStatus	Only specific to business document type (PO Response, PO Change, etc.)
lineNote	lineNote
projectRef	Managed through the contractRef at document level.
ReferencedLineID	Only specific to business document type (PO Response, PO Change, etc.)
QuantifiedLine	Line
lineQuantity	lineQuantity
PricedLine	PricedLine
	NOT AN "EMPTY" CLASS (described below)
UnPricedLine	ItemLine
SummaryLine	Not managed
Not managed	CurrencyLine

OGC	IDA
ItemInstance	ItemInstance
instanceSerialNumber	instanceSerialNumber
instanceBatchNumber	instanceBatchNumber
instanceSupplierReference	instanceSupplierReference
instanceRegistrationNumber	instanceRegistrationNumber
instanceManufactureDate	instanceManufactureDate
instanceRegistrationDate	instanceRegistrationDate
instanceExpiryDate	instanceExpiryDate
Aspect	Aspect
aspectName	aspectName
aspectValue	aspectValue
aspectValueUnit	aspectValueUnit
Item	Item
itemName	itemName
unitOfMeasure	unitOfMeasure
itemDescription	itemDescription
sellerItem	sellerItem
Gtin	gtin
itemUuid	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
ExtendedItemID	
extendedItemID	Not managed
extendedItemIDSource	
PricedItem	PricedItem
unitPrice	PricedItem.unitPrice
commodityClass	Item.commodityClass
PricedItemValidity	
quantityFrom	Not managed
quantityTo	
priceValidFrom	
priceValidTo	
FINANCIAL PACKAGE	
LineValue	PricedLine
totalLineValue	totalLineValue
PricedLineValue	PricedLine
extendedNetLineValue	netLineValue
LineTax	LineTax
lineTax	lineTax
LineChargeDiscount	PricedLine

OGC	IDA
lineChargeDiscount	chargesLineValue
	discountsLineValue
DocumentChargeDiscount	PricedDocument
documentChargeDiscount	chargesTotal
	discountsTotal
ChargeDiscountData	Not managed (only totals are managed, see above)
chargeDiscountType	
chargeDiscountPerCent	
chargeDiscountName	
DocumentTotals	PricedDocument
DocumentTotals	DocumentTotal
PricedDocumentTotals	PricedDocument
subTotal	netDocumentTotal
chargesTotal	chargesTotal
discountsTotal	discountsTotal
StandardTaxData	StandardTaxData
taxtype	taxtype
taxBand	taxBand
taxPerCent	taxPerCent
TaxTypeTotal	TaxTypeTotal
taxTypeTotal	taxTypeTotal
ExchangeCurrency	ExchangeCurrency
targetCurrencyCode	targetCurrencyCode
baseCurrencyCode	baseCurrencyCode
targetToBaseRate	targetToBaseRate
exchangeRateDate	exchangeRateDate
exchangeRateSource	exchangeRateSource
PARTY PACKAGE	
Party	Party & ContactInformation
organisationName	Party..organisationName
contactName	ContactInformation..contactName
partyNote	Party.partyNote
partyUuid	Not managed (cf. § 8. Appendix 1: Issues and decisions made, page 63, issue #BDC6)
Address	Address
postCode	postCode
countryCode	countryCode
UnstruturedAddress	Address
addressLine	addressLine

OGC	IDA
StructuredPostalAddress	Address
streetDescription	streetDescription
locality	Only one attribute (cityName) for the two attributes locality and town.
town	cityName
administrativeArea	Not managed as a specific attribute but the <i>addressLine</i> attribute can be used for all extra-information on an address.
primaryAddressableObjectName	
secondaryAddressableObjectName	
ukinternalCode	
Not managed	postBox
Not managed	region
ElectronicAddress	ContactInformation
directLine	directLine
switchboard	switchboard
fax	fax
mobilePhone	mobilePhone
eMail	eMail
ExtendedElectronicAddress	Not managed
electronicAddress	
eAddressChannel	
eAddressUse	
eAddressTimeValidity	
PersonStructuredName	ContactInformation
title	Managed as a single attribute (contactName)
forename	
surname	
nameSuffix	
jobTitle	jobTitle
department	department
knownAs	
OrganisationIdentifier	Party
dunsNumber	dunsNumber
gln	Gln
taxIdentifier	taxNumber
registrationNumber	registrationNumber
registrationIn	registrationCountryCode
OrganisationDetail	Not managed. The Party.partyNote attribute can be used in order to specify any extra information on a party.
tradingAs	
parentOrganisation	
registeredName	
registeredAddress	
PurchaseCard	PurchaseCard

OGC	IDA
CardNumber	CardNumber
CardHolderName	CardHolderName
ExpiryMonth	ExpiryMonth
ValidFromMonth	ValidFromMonth
VerificationValue	VerificationValue
IssueNumber	IssueNumber
CRI	CRI
EFTaddress	EFTaddress
AccountNumber	AccountNumber
BranchID	BranchID
BankName	BankName
AccountName	AccountName
SWIFTCode	SWIFTCode
SalesPoint	SalesPoint
buyerRefForSeller	Not managed
AccountsReceivable	AccountsReceivable
buyerRefForSeller	Not managed
OrderPoint	OrderPoint
sellerRefForBuyer	Not managed
DeliveryPoint	DeliveryPoint
InvoiceTo	InvoiceTo
CustomerService	CustomerService
DespatchPoint	DespatchPoint
Carrier	Carrier
Factor	Not managed
Originator	Not managed
PurchasingManager	Not managed
PurchaseOrder	PurchaseOrder
requInvCurrency	invoiceCurrency
PurchaseOrderID	Managed as a single attribute (Document.documentID)
PurchaseOrderLine	PurchaseOrderLine
requiredByDate	requiredByDate
deliveryTerms	deliveryTerms

OGC	IDA
Not managed	maxBackOrderQuantity
INVOICE	
Invoice	Invoice
	Not managed. Note that such information might not be necessary since a CreditNote is a separate business document in the IDA model.
invoiceType	
taxPointDate	Managed at line level (Invoice->InvoiceLine.receiptDate)
taxTotal	taxTotal
cardAuthorisation	Not managed
InvoiceLine	InvoiceLine & PurchaseOrderLine
costCenterRef	Not managed
costCodeRef	Only managed in the PurchaseOrder document
Not managed	receiptDate
	paymentExpectedDate
Managed in a dedicated class	poLineID
	purchaseOrderID
Managed differently (see below)	poResponseLineID
	poResponseID
	poChangeLineID
	poChangeID
	ReceiptAdviceLineID
	ReceiptAdviceID
POLineNumber	Managed differently (not in a dedicated class): poLineID + purchaseOrderID
PurchaseOrderID	Managed differently (see above).

8 APPENDIX 1: ISSUES AND DECISIONS MADE

This section gives the different issues that arose during the modelling of business documents together the associated decision.

The status of an issue can be one of the following:

- *OK*: a decision that we think is satisfying was taken;
- *Unresolved yet*: no decision was taken yet;
- *Open*: a decision was taken but we feel that it has almost as many cons as pros.

#	Issue description	Status	Decision
General			
GEN1	<i>Why not use an existing standard rather than develop a new model?</i>	Open	<p>The main reasons why it was decided to propose a new model are the following:</p> <ul style="list-style-type: none"> • There are numerous existing standards covering the e-ordering and e-invoicing process; but most existing models, such as Rosetta.net and OAGIS are very complex and difficult to implement; • Work was already carried out on the subject by other Member states providing us with a solid base for describing a core set of requirements; • UBL, seen by many as one of the most promising standard in the domain, was in its version 0.7 at the time we started our work. <p>Furthermore, the IDA model could also be considered as the description of a common denominator for the requirements in public e-procurement in Europe. Apart from its XML implementation, it could also:</p> <ul style="list-style-type: none"> • Be used as a common basis for mapping different existing standards thus facilitating interoperability between them (each particular standard only has to provide a converter to the IDA model rather than to all other standards); • Be seen as a contribution to enrich other existing standard in particular UBL.
UML modelling			
UM1	<i>Use of inheritance?</i>	OK	Inheritance should be used when the generalisation of two or more subclasses and its parent class appears natural and that it facilitates the comprehension of the model.
UM2	<i>Use of packages?</i> Is it useful to split classes from the information model between	OK	We used packages in order to structure the classes constituting our information model and its

#	Issue description	Status	Decision
	packages? What should be the logic of this separation?		documentation.
	Business documents content/structure		
BDC1	<i>Is it desirable that the content of a business document be redundant?</i> For example should price totals that could be calculated automatically from detailed price be included creating redundancy in the message.	OK	Redundant information can be added when necessary: <ul style="list-style-type: none"> • It can make information as explicit as possible; • Redundancy has very few cons.
BDC2	<i>Should documents be self-contained?</i> For example when accepting an order (<i>Order Response</i> business document) should the detail of the Order be repeated in the response?	OK	Business documents should be, as much as possible, self-contained: <ul style="list-style-type: none"> • Repeating information does not make the process significantly more complex or time-consuming; • This makes the agreement between trading partners less ambiguous/more explicit; • This can avoid having to consolidate several business documents in order to check or understand the information contained in one document; this is particularly important when documents serve as a proof for non-repudiation and/or they are archived as separate files.
BDC3	<i>How should references between parts of the same or different documents be modelled?</i>	OK	We decided not to use references from one part to another one in the same document: <ul style="list-style-type: none"> • As most business documents are data-centric¹⁰, the most frequent case where internal references are needed is to avoid repeating the same information several times; we believe that the price for repeating information in the same business document is not very high (the difference in size is low); • On the contrary, generating and handling internal references is relatively difficult technically. <p>For references between different documents, this can be done at document and line level.</p>
BDC4	<i>How could we lower the cost of implementation for SMEs?</i>	1/ OK. 2/ Unresolved yet.	1/ The following rules should tend to reduce the cost of adoption in general: <ul style="list-style-type: none"> • The creation of business documents should not require complex or expensive software; the simplest scenario consisting of the use of a spreadsheet/word processor + a macro should be possible; • The handling of a business document should not require complex and expensive software; business

¹⁰ As opposed to text-centric documents in which cross-references are common practice.

#	Issue description	Status	Decision
			<p>documents should be viewable in an Internet browser¹¹; this also explains BDC2;</p> <ul style="list-style-type: none"> The transmission of business documents should not require specific software and should be independent from any transport protocol; the simple scenario consisting of using email should be possible. <p>2/ Another possible option is to develop different levels of agreements defining processes and business documents (more or less complex) for each level.</p>
BDC5	<i>Should the IDA e-procurement model should remain simple or should it be as complete as possible?</i>	Open	We opted for a compromise between a simple model and a complete model covering all possible cases, i.e. extensible for big companies and manageable for SMEs.
BDC6	<i>Should we use UUID¹²?</i> UUID provide the simplest mechanism, at machine level, to generate identifiers for business documents that are guaranteed to be unique.	Open	<p>We decided not to:</p> <ul style="list-style-type: none"> They need to be generated automatically, which contradicts decision BDC4.
BDC7	<i>Use XHTML as a structured text type?</i>	OK	<p>For the time being, we made the decision not to:</p> <ul style="list-style-type: none"> Although generating XHTML text can be relatively easy (one can always generate plain text or simply a series of paragraphs marked by <code>p</code> elements), receiving and handling information that might be structured with XHTML is significantly more difficult.
BDC8	<i>Use of UBL core components/representation terms?</i>	Open	<p>We have decided to conform when possible some to the Core components data types with the same definition. Whether the IDA XML schemas will use their XML schemas definition (making our model dependent from theirs) is not decided yet.</p>
BDC9	<i>Structured vs. unstructured addresses?</i> UBL defines a very structured address component with many specific fields (floor, room, buildingName, buildingNumber, etc.) whereas Rosettanet and eHandel define an unstructured version of an address composed mainly of Country, ZipCode, PostBox and a series of address lines. In OGC, an address can be either structured or unstructured.	OK	<p>The IDA e-procurement Address component is a happy medium:</p> <ul style="list-style-type: none"> The great variety of the form an address can take (even within the same country) proves that it is always necessary to have free lines of text (i.e. it is possible not to structure an address by way of a list of specific fields); Pieces of information might still be processed automatically and are

¹¹ This will necessitate in a second phase to, similarly to UBL and other recent XML-based standards, the development of one or more XSLT stylesheets for each business document type.

¹² UUID stands for a Universal Unique IDentifier. These are 128 bit numbers assigned to any object which is guaranteed to be unique. The mechanism used to guarantee that UUIDs are Unique is through combinations of hardware addresses, time stamps and random seeds

#	Issue description	Status	Decision
			specified as specific fields.
BDC10	<i>Commodity codes for items?</i> OGC and eHandel specify one commodity code element corresponding explicitly to UNSPSC. Should the choice of the classification not be left to users (for example e-cl@ss or CPV)? Shouldn't there be several commodity codes elements to allow users to specify different codes for the same item?	OK	For the sake of simplicity, we propose to have only one commodity code but the corresponding attribute can refer (code datatype) to any classification standard.
BDC11	<i>Structured person name?</i> Is it necessary to allow for the decomposition of a person's name (forename, middle name, surname, title, etc.).	OK	Although having a structured person name can be a benefit since it allows for more precision, we have not found a real case where distinguishing the different components of a name was a real necessity. Therefore, for the sake of simplicity, a person name in the IDA model is a simple free text attribute.
BDC12	<i>Transport and packaging information?</i> A lot of information related to the transport and packaging of goods are important especially in the process of delivery. Should it be taken into account in the model?	Open	Logistics and transport can be complex field. Although there is a requirement for exchanging information related to transport and packaging between buyer and seller, we could not find a simple way to model it (in particular because we did not want to break the structure of lines on which all business documents rely and on which the simplicity and homogeneity of the whole model relies). Furthermore, transport and packaging information can always be provided in corresponding business documents (DespatchAdvice in particular) free-text attributes ¹³ (documentNote, lineNote, partyNote).
BDC13	<i>Should there be, on top of the lineNote generic attribute a specific attribute for comments in response documents</i>	OK	A specific "note" attribute has been added to <i>POResponse</i> , <i>POChange</i> and <i>InvoiceResponse</i> at line level so that a comment could be added, without deleting the comment present in the original message. For example, a seller may wish to reply to a PurchaseOrder and add a comment on a particular line for which the buyer has already specified a comment using <i>lineNote</i> . The seller can use the <i>poResponseLineComment</i> attribute without having to modify the original content of the <i>lineNote</i> attribute.
BDC14	<i>Language attribute</i> Should a language attribute be added to all text attributes to allow specifying the language they are expressed in?	Open	As far e-ordering and e-invoicing are concerned, very little free text is generally used. Nevertheless, all textual attribute (attributes of type <i>Text</i>) have a dedicated <i>language</i> attribute.
BDC15	<i>Contract reference</i> All business messages may (and usually does) refer to a specific contract. How much information on that contract should be repeated in each message?	Open	For the sake of simplicity, we propose to only specify the contract reference (but not information like <i>start date</i> and <i>end date</i>)

¹³ cf. 4. Extending the IDA e-procurement model, page 42.

#	Issue description	Status	Decision
BDC16	<p><i>Currency information</i></p> <p>The whole cycle from ordering through to payment could involve many different currencies. How can different currencies be managed while keeping the model relatively simple?</p>	Open	First, all attributes containing an amount has a <i>currency</i> attribute attached to it (cf. datatype <i>Amount</i>). As a general rule and in order to keep the business document's structure simple, when amounts are to be given in two different currencies, this is done at document level (only for totals) and not at line level.
XML			
XML1	<p><i>Use of attributes vs. elements</i></p>	Open	<p>Elements should be the main holders of content. Attributes should be used to hold metadata or characteristics of element content.</p> <p>We think that we should use attributes only for defining data types. In that case, attributes are used as qualifiers (such as <i>currencyCode</i> for data type <i>Amount</i> or <i>unitOfMeasure</i> for data type <i>Quantity</i>).</p>
XML2	<p><i>Use of namespaces</i></p>	Open	<p>There will be one namespace per package:</p> <ul style="list-style-type: none"> • Not using namespaces reduce interoperability; • Using too many namespaces makes the XML schemas more difficult to use/implement.
XML3	<p><i>What extension mechanism should be used?</i></p>	Open	See the <i>XML schema design guidelines</i> document.
XML4	<p><i>How should be files attached to messages? Should we allow for inline binary files or just a simple URI?</i></p> <p>There are many scenarios where attaching a file, either to the business document or a specific line, is a requirement or a real benefit. There is mainly two ways to include binary data in an XML message.</p> <p>The first one is to have inline binary data encoded in base64 in a dedicated XML element.</p> <p>The second one is to specify a URI in the XML document identifying the location of the attached file. In practice this may be a URI pointing to a document located somewhere on the Internet and accessible by a Internet browser. Another common practice is to use MIME as a packaging mechanism to embed in a single message different files.</p>	Open	<p>The first solution which consists in having inline binary data has several drawbacks. First, base64-encoding files makes them one-third larger. Second, this solution may lead to extremely large XML files that will be more time and memory-consuming to process if you just want to interpret the XML content. Again for simplicity purposes, we decided that a URI would be the only attribute of the <i>Attachment</i> object.</p>

9 APPENDIX 2: UML TO XML SCHEMAS CONVERSION RULES

R1/ file and directory structure:

There is one schema file per package except for the Document package where there will be one common schema (document.xsd) and one schema per business document class

There is one directory per package with the same name as the package (document, line, item,...)

R2/ Classes:

For each class that does not correspond to a business document (top-level class) of the information model:

=> a complex type of name: class name + 'Type'

For each class that correspond to a business document (PurchaseOrder, Invoice, POResponse, etc.):

=> a complex type of name: class name + 'Type' is defined

=> an element with the same name as the class referencing that comlexType is also defined

R3/ Associations:

=> When the association is of cardinality 0..1 or 1..1, a specific element is defined with the same name as the target class; the type of this element is the type corresponding to the target class;

=> When the association is of cardinality 0..* or 1..* a specific repetitive XML element is defined with the same name as the target class; the type of this element is the type corresponding to the target class.

R4/ Attributes:

=> A local element is defined with the same name as the attribute; the XML element type is the class that corresponds to its type as defined in the UML diagram.

R5/ Digital signature:

For the digital signature attached to business document an optional element conforming to the W3C xml dsig schema was added to the Document class.

10 APPENDIX 3: XML SCHEMAS DESIGN GUIDELINES

10.1 NAMING CONVENTIONS

RNM1. In order to conform to common practice as regards XML schemas, elements names **MUST** use “upper camel case”: **MUST** start with a capital. When a name is composed of several words:

- they should not be separated by an underscore (`_`) character;
- each word starts with a capital letter.

In order to conform to ebXML conventions, attribute names should use lower camel case (i.e. starts with a lower-case initial).

Example:

```
<ItemDescription versionId='v3'>...</ItemDescription>
```

RNM2. Underscores (`_`), dots (`.`) and dashes (`-`) **MUST NOT** be used in component names.

10.2 NAMESPACE

RNS1. An XML schema should have a target namespace.

RNS2. A particular namespace should correspond to a coherent collection of schemas or components.

10.3 VERSIONING

RVER1. The version of an XML schema is indicated, in accordance with W3C practice, in the `version` attribute of the `xsd:schema` element.

RVER2. The version **MUST** not be part of the namespace URI.

10.4 USE OF DATA TYPES DEFINITION

RDT1.

A component **MUST** be defined as a data type if:

- it can be used in different contexts with different element names or
- it has or it is foreseen to have other data types derived from it.

In all other cases, a component can be defined as a simple element.

10.5 USE OF ATTRIBUTES

RAT1. Elements should be the main holders of content. Attributes should be used to hold metadata or characteristics of element content.

10.6 MULTILINGUAL CONTENT

Although one should use when possible codified values that would be independent from the language, there are many cases where the value of a component is textual. It therefore depends on the language and can also be given in different languages (for example for textual description of items of a catalogue).

RMC1. Components that can have a textual value **SHOULD** be characterised by an `xml:lang` attribute.

RM C2. Components that can have a textual value SHOULD, when applicable, be made repetitive (cardinality 0..* or 1..*) and hold a language attribute (xml:lang) so that its value can be produced in several languages.

Example:

```
<Item>
...
  <Description xml:lang="en">Video Camera</Description>
  <Description xml:lang="fr">Caméra Vidéo</Description>
  <Description xml:lang="es">Viodeocámara</Description>
...
</Item>
```

10.7 EXTENSION MECHANISM

REM1. In order to design extensible components, we suggest using inheritance/specialisation of complexTypes.

10.8 SCHEMA DOCUMENTATION

RSD1. All components should have a corresponding documentation (in English) using the xsd:annotation element.

10.9 elementFormDefault AND attributeFormDefault

RFD1. elementFormDefault MUST be set to qualified and attributeFormDefault SHOULD be set to unqualified.

Example:

```
<xsd:schema
  targetNamespace="http://www.europa.int/IDA/PurchaseOrder"
  xmlns="http://www.europa.int/IDA/PurchaseOrder"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  version="1.0">
```

11 APPENDIX 4: ACRONYMS

Acronym	Description
BACS (Banker's Automated Clearing System)	Electronic payment method. (http://www.bacs.co.uk/home/home.php)
B2B (Business to Business)	B2B is e-commerce transaction between two companies.
CEN (European Committee of Standardization)	The CEN is a European standardisation body contributing to the objectives of the European Union and European Economic Area with voluntary technical standards. (http://www.cenorm.be)
CPV (Common Procurement Vocabulary)	Standard codification for use in public procurement. (http://simap.eu.int/EN/pub/src/welcome.htm)
CRI (Customer Reference Identifier)	Unique company registration identifier.
DTD (Document Type Definition)	Specific definition that follows the rules of the Standard Generalized Markup Language (SGML). A DTD is a specification that accompanies a document and identifies what the codes (or mark-up) are that separate paragraphs, identify topic headings, and so forth and how each is to be processed.
DUNS	DUNS stands for "Data Universal Numbering System." It is a unique nine-digit numbering system that is used to identify a business.
EAN	Non-profit, business-led association that manages a worldwide system that enables the identification of items, trade and logistic units, services and locations. The purpose is to provide a common language for international trade and commerce that is applicable to virtually all industrial and commercial sectors. The system includes standards that cover; identification and supplementary information, bar code specifications, and an internationally used subset of the <i>EDIFACT</i> <i>EDI</i> messages called EANCOM. (http://www.ean-int.org/)
EbMS	ebXML Messaging Infrastructure. Messaging framework on which ebXML is based. (http://www.ebxml.org)
EbXML	Electronic Business eXtensible Markup Language (http://www.ebxml.org)
eCl@ss	Classification standard (http://www.eclass-online.com/)
EFT	Electronic Funds Transfer.
GLN	The GLN (Global Location Number) provides a standard (EAN/UCC) means to identify legal entities, trading parties and locations to support the requirements of electronic commerce.
GTIN	The GTIN (Global Trade Item Number) is the foundation for the EAN.UCC System for uniquely identifying trade item (products and services).
HTTP	Hypertext Transfer Protocol used for the Web.
IDA	Interchange of Data between Administrations
INCOTERM (International Commercial Terms)	Standard trade definitions that are most commonly used in international contracts. (http://www.iccwbo.org/index_incoterms.asp)
ISO	International Organization for Standardization (http://www.iso.ch)

OASIS	Organization for the Advancement of Structured Information Standards. not-for-profit, global consortium that drives the development, convergence, and adoption of e-business standards. (http://www.oasis-open.org/)
OGC	The Office of Government Commerce (OGC) is an independent Office of the Treasury (UK) reporting to the Chief Secretary. It is responsible for a wide-ranging programme which focuses on improving the efficiency and effectiveness of central civil Government procurement. (http://www.ogc.gov.uk/)
PDF	Adobe® Portable Document Format (http://www.adobe.com)
PO	Purchase Order
RFC	Request For Comment. Set of standard specifications related to the Internet. (http://www.ietf.org/rfc.html)
Rosettanet PIPS (Rosettanet Partner Interface Processes)	Set of process and message definition for e-commerce. (http://www.rosettanet.org)
SMEs	Small and Medium Enterprises
SMTP	Simple Mail Transfer Protocol. Standard protocol for delivering emails over the Internet.
SOAP	Simple Object Access Protocol. Protocol allowing transport of XML information on which Web Services are based. (http://www.w3.org/2000/xp/Group/)
SWIFT (Society of Worldwide Interbank Financial Telecommunications)	SWIFT is the industry-owned cooperative supplying secure, standardised messaging services and interface software to 7,500 financial institutions in 200 countries. (http://www.swift.com)
UBL	Universal Business Language. Standard library of XML business documents (purchase orders, invoices, etc.) develop by OASIS. (http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ubl)
UCC (Uniform Code Council)	Global, multi-sectoral standards for supply-chain efficiency in electronic environment. (www.uc-council.org)
UML (Unified Modeling Language http://www.omg.org)	Standard notation for object-oriented software design.
UN/EDIFACT	United Nations / Electronic Data Interchange for Administration, Commerce and Transport
UNSPSC	United Nations Standard Products and Services Code. Classification standard for product.
URI (Unified Resource Identifier)	Way to identify any of those points of content, whether it be a page of text, a video or sound clip, a still or animated image, or a program.
URL (Uniform Resource Locator)	Most common form of URI (Web page address), which is a particular form or subset of URI called a Uniform Resource Locator (URL).
VAT	Value Added Tax
X12	ANSI standard for the exchange of business transaction information
xCBL	XML Common Business Library. Royalty-free XML-based B2B standard originally developed by <i>Commerce One</i> . (http://www.xcbl.org/)

XML	EXtensible Markup Language. (http://www.w3.org/XML/)
W3C (World Wide Web Consortium)	The World Wide Web Consortium (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential. (http://www.W3.org)